前言

出版说明

自从方倍工作室在博客园推出微信公众平台开发系列教程后，受到广大微信开发人员及爱好者的热情关注，相关文章的日访问量高达2万人次，而《微信公众平台开发入门教程》的阅读量超过50万，博客访问量总计超过500万，成为微信公众平台开发更新最快，传播最广、受众最多、资料最全的博客，很多博文被许多有影响力的网站转载，并被各大搜索引擎收录且排名靠前，这些是我们始料未及的。

然而更让人高兴的是，很多开发者通过学习我们的教程掌握了微信公众平台开发技术，并且通过微信开发赚到了钱。2013年10月17日，我们在QQ空间发布了新版《微信公众平台开发入门教程》链接后，网友“我叫不熬夜？☺”在空间中回复，他之前通过学习我们的微信开发教程赚到了2000元，而他目前还只是一名学生。这条回复记录至今还保存在方倍工作室的QQ空间中，给了我们不断前进的动力。

为了推出更好、更有价值的作品，我们整合已有的教程资源，并从2013年到2014年的几百个开发案例中挑选出最受欢迎的功能应用，编写了本书。书中全面介绍微信公众平台（包括微信支付在内）的所有接口及使用方法，并且辅以30多个功能应用案例及技巧，同时在分析过程中融合相关知识与技术，力求使读者不但“知其然”，而且“知其所以然”，以期为读者奉献一本含金量高的书籍。

读者对象

本书适合以下人群：

· 想了解移动互联网及微信公众平台发展的行业从业人员。

· 想了解微信公众平台产品使用方法、技巧及效果评估的微信营销人员。

· 想提高会员活跃度、提高指标转化率、推进品牌推广的公众平台运营人员。

· 想学习微信公众平台开发的技术人员。

· 想使用微信公众平台兼职开发、创业等渴望更成功的人员。

· 想搭建企业内部强大及实用的微信公众平台的开发团队。

阅读指南

本书分为10章。

第1章简要介绍微信及其三大平台：微信公众平台、微信开放平台和微信智能开放平台，重点介绍微信公众平台的注册方法及各项功能。

第2章介绍申请新浪SAE作为免费服务器资源的方法，同时介绍如何配置微信公众平台的开发者中心以及启用过程中常见问题的解决方法，最后介绍如何使用微信调试器及接口调试工具进行开发调试。

第3章介绍微信公众平台基础接口的三个部分，包括接收用户发送的六种消息、以六种方式发送被动响应消息、接收关注及取消关注的事件推送消息。所有消息类型都给出了PHP语言的实现方法。这些消息类型是微信公众平台与用户进行交互的基础功能。

第4章介绍自定义菜单与高级接口的九大功能。这些功能都需要额外向微信公众平台申请权限。它使微信交互不再局限于消息交互，而是能达到更广、更深、更有想象力的层次。基于这些接口的功能也将是今后微信开发的主要趋势，本书对这些功能都给出了PHP的实现代码。

第5章详细介绍微信支付的申请流程及功能，并且对微信支付中的两种主要支付方式的实现原理进行深入剖析，另外还讲解商户功能的其他相关接口的内容。

第6章详细介绍微信小店的搭建方法及二次开发的方法，更进一步丰富和完善微信小店的功能。

第7章介绍八个生活类应用的开发。书中对这些应用从接口申请，数据获取直到开发实现，案例截图，都进行了详细描述。最后介绍了如何将这些功能进行整合。读者掌握这些应用的开发以后，可以丰富自己公众账号的功能。

第8章介绍八个娱乐类应用的开发。这些应用都是非常受用户欢迎的，读者学习完后可以快速移植到自己的微信公众平台，提高粉丝存在的价值。

第9章介绍企业最常用功能的开发，其中包括部分高级接口在企业应用的实现。这些功能为企业开发提供了方向及技术指导。

第10章介绍微信开发有关的其他实用技巧以及微信JS接口。这些功能可以为公众账号用户带来更有趣的体验。

本书程序案例采用广泛流行的PHP、MySQL、XML、CSS、JavaScript、HTML5等程序语言及数据库实现，所有案例均在书中给出了核心实现代码并做了详细讲解。初学者可以在了解PHP和MySQL语法之后，从头至尾学习，对于其中难以理解的部分可以查阅相关资料。对于企业应用类的开发还需要具有一定的JavaScript、CSS等语言基础。有经验的微信公众平台开发人员可以根据自己的需要直接切入相应章节，其他从业人员则可以选择自己感兴趣的内容阅读。

由于作者水平及能力有限，加之时间仓促，书中难免出现错误和不妥之处，对于一些依赖第三方功能的实现也可能由于外部原因难以保证可以永久使用，恳请读者批评指正，并欢迎探讨，一并感谢！

源码及勘误说明

本书的源代码可以从华章网站本书页面下载，网址为：http://www.hzbook.com/。

勘误说明及源码更新可以从方倍工作室的微信公众账号（账号：PondBayStudio）下载，欢迎关注并索取更新。此外，我们也会经常发布微信最新动态及技术信息。

致谢

首先感谢“微信之父”张小龙先生及微信团队，是他们创造了微信这一经典传世之作。

感谢机械工业出版社华章公司王彬编辑的支持，促成了这本书的改版。

感谢我最亲爱的家人在背后的默默支持与付出。

本书成书过程中，也得到了诸多同行的支持与鼓励，在此一并致谢。

谨以此书献给所有热爱移动互联网和微信及微信公众平台的人们。

方倍

2015年1月于深圳

第1章　微信公众平台介绍

微信自推出后，便如星火燎原之势发展，截至2014年7月，微信用户数已近8亿，月活跃用户接近4亿，公众账号达580万个，每日新增数为1.5万，是亚洲地区用户群体最大的移动端即时通信软件，同时也被公认为目前唯一拿到移动互联网门票的应用。

本章主要介绍微信公众平台的注册及使用。

1.1　微信及其三大平台

微信（英文名：WeChat）是腾讯公司于2011年年初推出的一款可以发送文字、表情、图片、语音、视频、位置、链接，并支持语音实时对讲的手机聊天软件。用户可以通过“添加QQ好友”、“添加手机联系人”、“摇一摇”、“附近的人”、“一起按”、“搜号码”、“查找公众账号”、“扫描二维码”等多种方式添加好友或关注微信公众账号，也可以将内容发送给好友以及分享到朋友圈。同时，微信还提供“微信支付”、“理财通”、“微信红包”、“嘀嘀打车”、“表情”、“游戏”等贴近生活的功能。

微信由腾讯广州研发中心产品团队开发，该团队经理张小龙被称为“微信之父”，公司总裁马化腾确定该产品的名称为“微信”。

微信的官方网站是：http://weixin.qq.com/。图1-1是微信图标。

图1-1　微信图标

2012年8月23日，腾讯公司推出微信公众平台，其宣传口号是“再小的个体，也有自己的品牌”。微信公众平台是微信公众账号所有者（政府、媒体、企业、组织或个人等）进行品牌推广、减少运营成本、提高影响力、与用户进行互动交流及提供服务的平台，公众账号通过消息、事件、菜单等交互方式为用户提供服务。例如：公众账号“招商银行信用卡中心”为持卡人提供信用卡绑定，查询信用卡账单、额度及积分；快速还款、申请账单分期；微信转接人工服务；信用卡消费，微信免费笔笔提醒等功能，同时还为非持卡人提供微信办卡功能。微信公众平台的官方网址是http://mp.weixin.qq.com/。微信公众平台还有国际版（也称海外版），其官方网址为：http://admin.wechat.com/。

除了公众平台以外，还有微信开放平台及微信智能开放平台。

微信开放平台是为移动应用开发者提供的内容分享接口，开发者可以在iOS、Android以及WP8平台上使用开放平台的SDK来开发分享功能，使用户可以在APP上分享内容给微信好友和微信朋友圈。微信开放平台的官方网址是http://open.weixin.qq.com/。

智能开放平台目前包括语音开放平台和图像开放平台。语音开放平台目前已开放的有通用语音识别、词表识别、语法识别、语音合成等语音技术。微信图像开放平台致力于为第三方应用提供免费的图像识别技术和服务。智能开放平台的地址为http://pr.weixin.qq.com/。

1.2　微信公众账号的注册

1.2.1　服务号/订阅号的注册

要使用微信公众平台，需要先注册一个微信公众平台账号。

在浏览器中输入微信公众平台的官方网址（http://mp.weixin.qq.com/），进入后如图1-2所示。

图1-2 微信公众平台首页

可以看到右上角有“第一次使用公众平台？立即注册”字样，单击“立即注册”链接，进入注册页面，如图1-3所示。

图1-3　基本信息界面

在基本信息界面中填写邮箱、密码、验证码，并勾选“我同意并遵守《微信公众平台服务协议》”，然后单击“注册”按钮，进入邮箱激活界面，如图1-4所示。

图1-4　邮箱激活界面

同时邮箱中将收到激活微信公众平台账号的确认邮件，如图1-5所示。

图1-5　激活邮件

单击邮箱中的链接，成功激活账号之后，注册页面自动跳转到信息登记界面。在界面中要求选择相应的运营主体是组织还是个人，其中组织类型又细分为政府、媒体、企业和其他组织。根据运营主体的不同，要求提供不同的资质材料及证明，如图1-6所示。

图1-6　信息登记界面

填写完信息登记界面后，再进入选择类型界面。公众账号类型可以选择服务号和订阅号，如图1-7所示。

图1-7　选择类型界面

服务号的特点包括：每月可群发一条信息给粉丝且群发的消息显示在聊天列表，下发消息即时通知粉丝，默认可以自动获得自定义菜单，可以申请微信认证获得高级接口权限，服务号旨在为用户提供服务。订阅号的特点包括：每天可群发一条消息给粉丝，群发消息隐藏在订阅号文件夹中，群发消息不会提示推送，认证后可申请自定义菜单，订阅号主要用于提供信息和资讯。

因此，企业和组织一般选择服务号，而媒体及个人一般选择订阅号。

选择好类型后，单击“继续”按钮，弹出公众账号信息界面，如图1-8所示。

图1-8　公众账号信息界面

填好信息之后，单击“完成”按钮，将得到成功创建公众账号的提示，如图1-9所示。

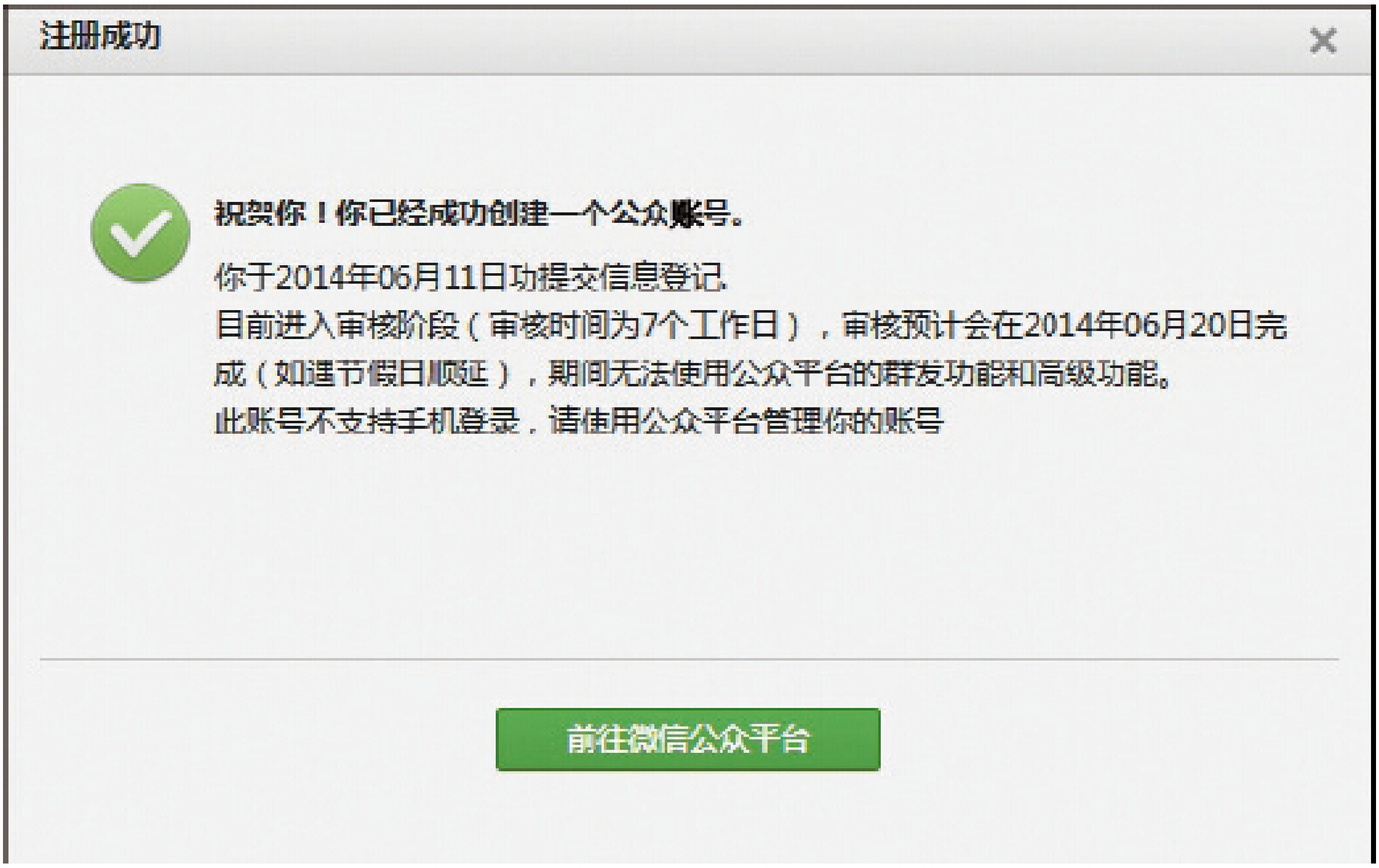


图1-9　注册成功界面

单击“前往微信公众平台”将进入账号信息。在“设置”→“公众账号设置”中可以查看到账号的基本信息，如图1-10所示。

图1-10　公众账号设置页面

1.2　微信公众账号的注册

1.2.1　服务号/订阅号的注册

要使用微信公众平台，需要先注册一个微信公众平台账号。

在浏览器中输入微信公众平台的官方网址（http://mp.weixin.qq.com/），进入后如图1-2所示。

图1-2　微信公众平台首页

可以看到右上角有“第一次使用公众平台？立即注册”字样，单击“立即注册”链接，进入注册页面，如图1-3所示。

图1-3　基本信息界面

在基本信息界面中填写邮箱、密码、验证码，并勾选“我同意并遵守《微信公众平台服务协议》”，然后单击“注册”按钮，进入邮箱激活界面，如图1-4所示。

图1-4　邮箱激活界面

同时邮箱中将收到激活微信公众平台账号的确认邮件，如图1-5所示。

图1-5　激活邮件

单击邮箱中的链接，成功激活账号之后，注册页面自动跳转到信息登记界面。在界面中要求选择相应的运营主体是组织还是个人，其中组织类型又细分为政府、媒体、企业和其他组织。根据运营主体的不同，要求提供不同的资质材料及证明，如图1-6所示。

图1-6　信息登记界面

填写完信息登记界面后，再进入选择类型界面。公众账号类型可以选择服务号和订阅号，如图1-7所示。

图1-7　选择类型界面

服务号的特点包括：每月可群发一条信息给粉丝且群发的消息显示在聊天列表，下发消息即时通知粉丝，默认可以自动获得自定义菜单，可以申请微信认证获得高级接口权限，服务号旨在为用户提供服务。订阅号的特点包括：每天可群发一条消息给粉丝，群发消息隐藏在订阅号文件夹中，群发消息不会提示推送，认证后可申请自定义菜单，订阅号主要用于提供信息和资讯。

因此，企业和组织一般选择服务号，而媒体及个人一般选择订阅号。

选择好类型后，单击“继续”按钮，弹出公众账号信息界面，如图1-8所示。

图1-8　公众账号信息界面

填好信息之后，单击“完成”按钮，将得到成功创建公众账号的提示，如图1-9所示。

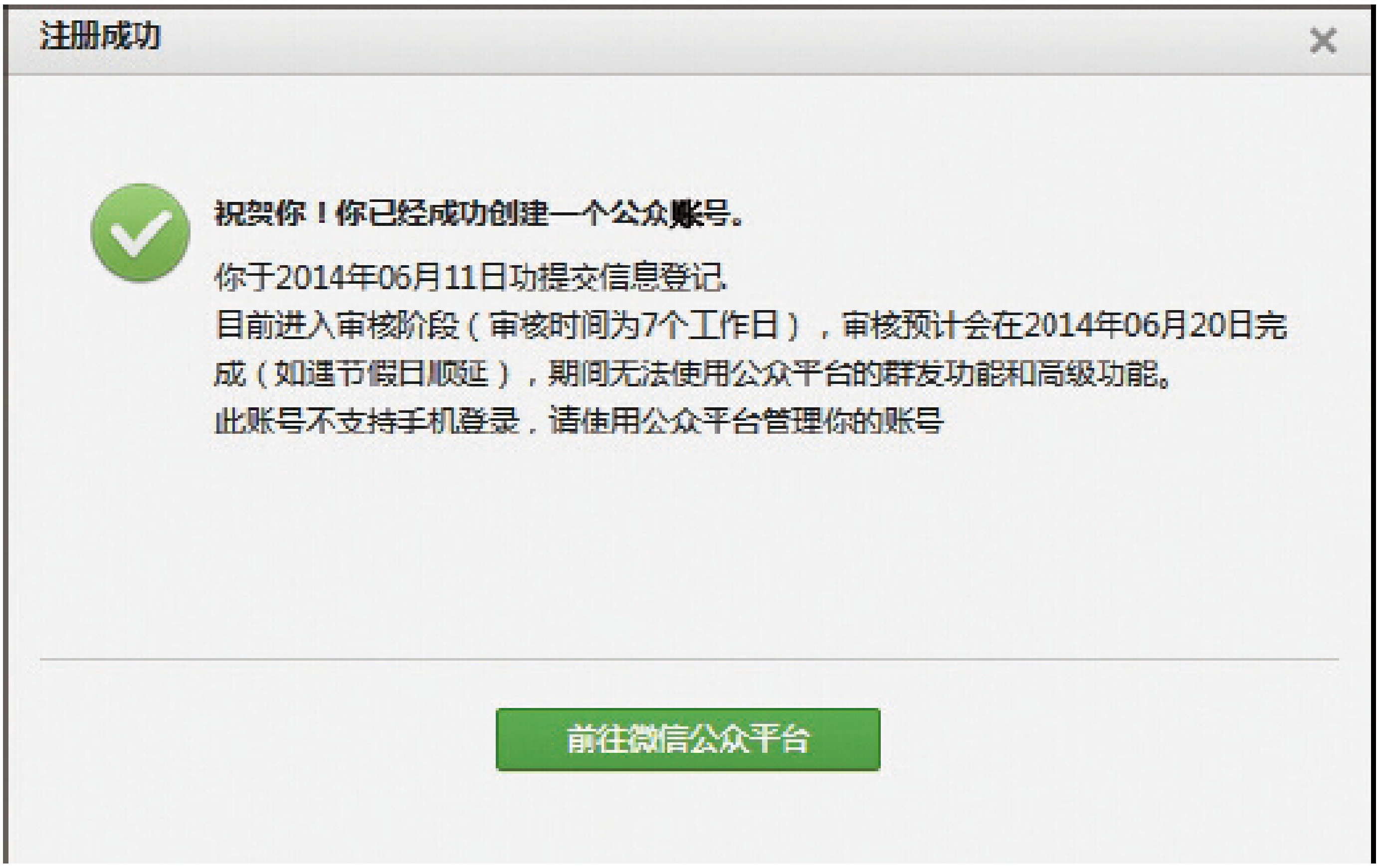


图1-9　注册成功界面

单击“前往微信公众平台”将进入账号信息。在“设置”→“公众账号设置”中可以查看到账号的基本信息，如图1-10所示。

图1-10　公众账号设置页面

1.2.2　测试号的注册

除了普通的服务号及订阅号之外，微信公众平台还为开发者提供了测试号的申请，开发者只需要用微信的“扫一扫”功能扫描二维码即可获得一个有效期限为一年的测试账号，该账号不需要认证即可拥有普通账号认证后才具有的权限。

注册微信测试号的地址是http://mp.weixin.qq.com/debug/cgi-bin/sandbox?t=sandbox/login。打开该链接后，页面如图1-11所示。

图1-11　微信公众平台接口测试账号申请

单击图1-11中的“登录”按钮，将跳转到下述链接http://open.weixin.qq.com/connect/qrconnect?appid=wx39c379788eb1286a&scope=snsapi_login&redirect_uri=http%3A%2F%2Fmp.weixin.qq.com%2Fdebug%2Fcgi-bin%2Fsandbox%3Ft%3Dsandbox%2Flogin。网页将显示微信扫描二维码登录页面，如图1-12所示。

图1-12　微信扫码二维码

用微信扫描图1-12中的二维码后，将弹出应用登录界面，如图1-13所示。

图1-13　应用登录

单击“确认登录”按钮后，将得到一个有效期限为一年的测试账号。该账号拥有appID和appsecret，并且可以对接口进行配置，如图1-14所示。

图1-14　管理测试号

同时，该账号拥有高级接口的权限，如图1-15所示。

体验接口权限表

接口	操作
接收用户消息	
向用户回复消息	
接受事件推送	
会话界面自定义菜单	
语音识别	关闭
客服接口	
OAuth2.0网页授权（仅关注者才能授权）	修改
生成带参数二维码	
获取用户地理位置	关闭
获取用户基本信息	
获取关注者列表	
用户分组接口	
上传下载多媒体文件	
高级群发接口	
设置用户备注名接口	
设备功能接口	关闭

图1-15　体验接口权限表

1.3　微信公众平台的使用

登录微信公众平台后，可以看到导航菜单、公众账号类型及名称等基本信息，如图1-16所示。

图1-16　微信公众平台后台

下面以微信公众账号“方倍工作室”为主要例子，介绍微信公众平台的各项菜单及使用。

1.3.1　功能

1.群发功能

群发功能是指微信公众平台最常用也最重要的功能之一。

根据需要，运营人员填写文字（或图片/语音/视频/图文等，需要先上传素材）内容后，选择群发对象、性别、群发地区发送即可。获得微信支付权限的公众账号还能群发商品信息。

群发功能界面如图1-17所示。

图1-17　新建群发消息

在“已发送”页面中，可以看到已经群发的消息，如图1-18所示。

图1-18　已发送的群发消息

2.自动回复

在编辑模式下，可以设置三种类型的自动回复。

· 被添加自动回复：当微信用户关注您的微信公众账号时自动推送的一条内容，支持文字、图片、语音、视频等类型。

· 消息自动回复：当微信用户发送消息给公众账号时，若未设置关键词自动回复或匹配不到相关的关键词，系统会自动推送该消息给粉丝。该类型信息1个小时内回复1~2条。

· 关键词自动回复：用户发送的消息内如果有已设置的关键词即可把设置在此规则名中回复的内容自动发送给用户。图1-19展示了关键词自动回复的设置方法。

图1-19　关键词自动回复

关键词自动回复有更详细的规则及使用方法，具体内容可以参考附录A。

通过设置上面三种类型的自动回复，可以完成一个全面的微信公众账号的内容回复功能。图1-20依次展示了三种自动回复的内容。

10:29
返回
方倍工作室
早上10:27
欢迎关注方倍工作室
你们地址在哪?
深圳市南山区高新南一道飞亚达科技大厦
谢谢
这是自动回复消息
发送

图1-20　自定义回复

3.自定义菜单

拥有自定义菜单权限的账号，也可以创建自定义菜单。该账号最多可以创建3个一级菜单，每个一级菜单下最多可创建5个二级菜单。每个菜单创建后需要为其设置响应动作。响应动作包括发送消息和跳转到网页。其中消息包括文字/图片/语音/视频/图文消息。图1-21展示了自定义菜单设置及其中一个子菜单回复图文消息的设置。

图1-21的自定义菜单发布后，效果如图1-22所示。

图1-21　自定义菜单设置

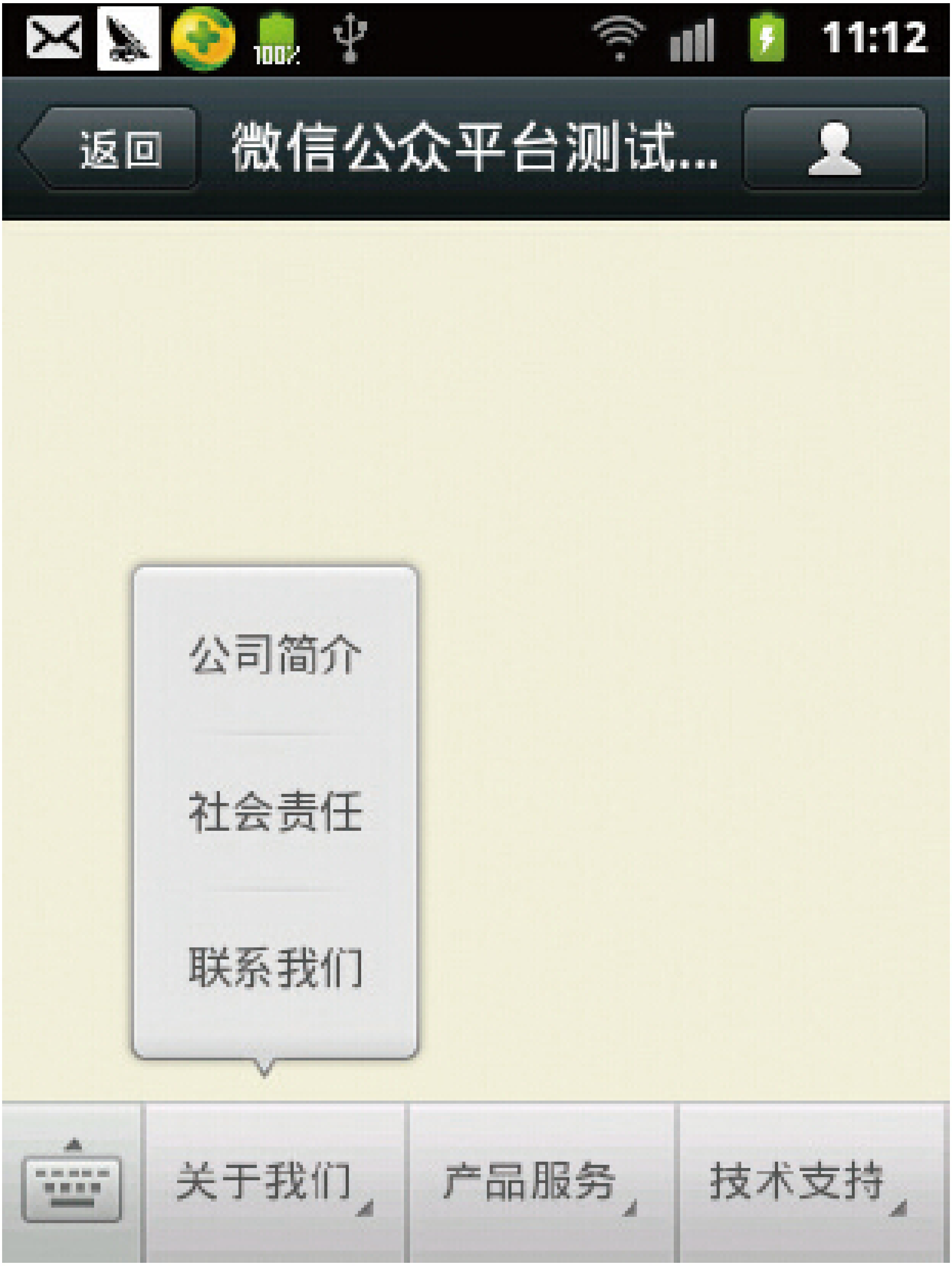


图1-22　自定义菜单效果图

4.其他功能

微信公众平台还有微信小店、多客服、模板消息、卡券功能、门店管理、设备功能等其他功能。运营者可以通过添加功能插件将其加入到功能列表中，而微信官方也在不断地推出新的功能插件供公众账号使用。图1-23是微信小店的图示。

图1-23　微信小店

1.3.2　微信支付

微信支付（商户功能），是公众平台向有出售物品需求的公众账号提供推广销售、支付收款、经营分析的整套解决方案。商户通过自定义菜单、关键字回复等方式向订阅用户推送商品消息，用户可在微信公众账号中完成选购支付的流程。商户也可以把商品网页转换为二维码，张贴在线下的场景，如车站和广告海报。用户扫描后可查看商品详情，在微信中直接购买。微信支付的后台界面如图1-24所示。

图1-24　微信支付

1.3.3　管理

1.消息管理

在消息管理中，可以查看全部消息（最近5天的消息），也可以查看今天、昨天、前天、更早以及星标的消息内容，另外还可以对消息内容进行搜索。鼠标移动到某条消息上，可以对其进行快捷回复或者单击星标收藏该消息，收藏后的消息会在后台永久保存。消息管理的后台界面如图1-25所示。

图1-25　消息管理

2.用户管理

在用户管理中，可以实现新建用户分组、移动用户至指定分组以及修改用户备注功能，如图1-26所示。鼠标移至用户头像上可以查看用户性别、地区、签名等信息。另外，被移至黑名单的用户将不能获得任何回复。

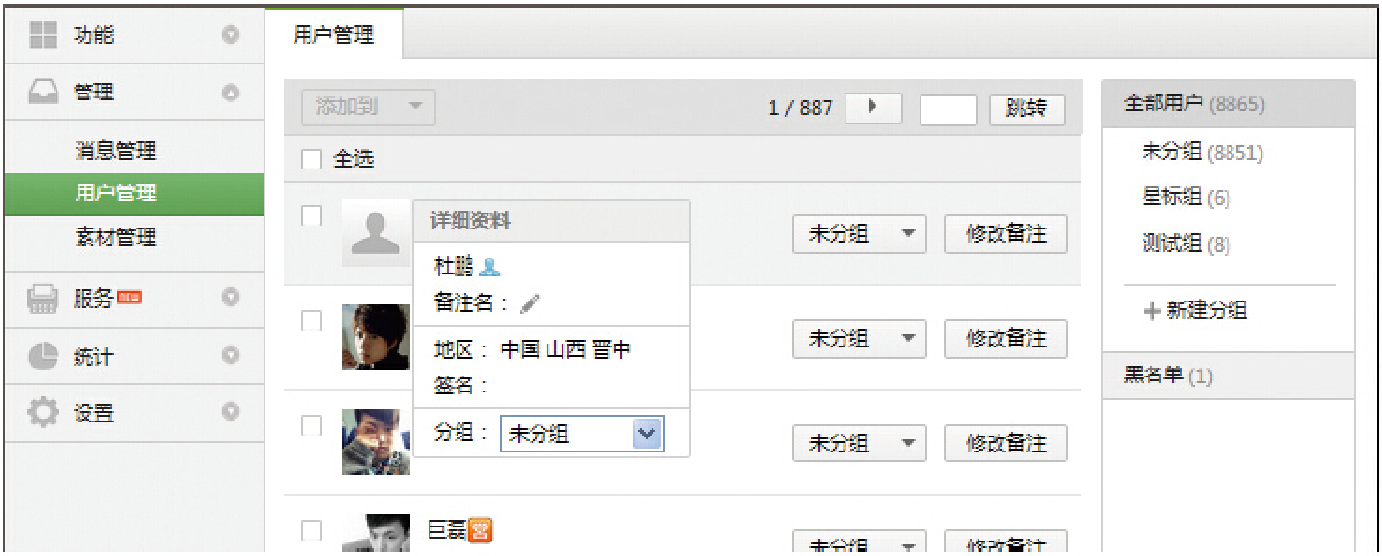


图1-26　用户管理

3.素材管理

在素材管理中保存了用户新建的图文消息、图片、语音及视频消息，如图1-27所示。这些信息可以用于自动回复，也能用于群发功能。

图1-27　素材管理

图文消息包括单图文消息以及多图文消息。图文消息包括以下几个部分：标题、作者（选填）、封面、摘要（仅单图文消息）、正文、原文链接（选填），如图1-28所示。其中多图文消息最多包含8条图文消息。而在开发模式下，多图文消息最多可以包含10条。

图1-28　图文消息

1.3.4　推广

微信公众平台推广功能是微信公众平台官方唯一的广告系统，公众账号运营者通过广告主功能可向不同性别、年龄、地区的微信用户精准推广自己的服务，获得潜在用户；也可通过流量主功能自愿将公众账号内指定位置分享给广告主做广告展示，按月获得收入。广告主功能如图1-29所示。

图1-29　广告主

1.3.5 统计

1.用户分析

用户分析主要分为用户增长和用户属性两大模块。用户增长模块按日、周、月为单位显示新关注人数，取消关注人数，净增关注人数，累积关注人数等几项指标的情况，如图1-30所示。用户属性模块中，可以根据性别、省份、城市、语言查看用户的分布情况。

图1-30　用户分析

2.图文分析

图文分析主要分为图文群发和图文统计两大模块。图文群发模块可以查看每篇图文消息的送达人数、图文页阅读人数/次数、原文页阅读人数/次数/转化率、分享转发人数/次数，如图1-31所示。而图文统计模块中，可以根据周期（日或小时）查看图文页阅读、原文页阅读、分享转发等指标情况。

送达人数	图文页阅读			原文页阅读			分享转发	
	人数	次数	图文转化率	人数	次数	原文转化率	人数	次数
8,778	2,719	3,821	30.98%	20	21	0.74%	229	282

图1-31　图文分析

3.消息分析

在消息分析中，可以根据周期（日或小时）查看消息发送人数、消息发送次数、人均发送次数等指标的情况，如图1-32所示。

图1-32　消息分析

4.接口分析

使用开发模式的公众账号，可以根据周期（日或小时）查看调用次数、失败率、平均耗时（毫秒）、最大耗时（毫秒）等指标的情况，如图1-33所示。

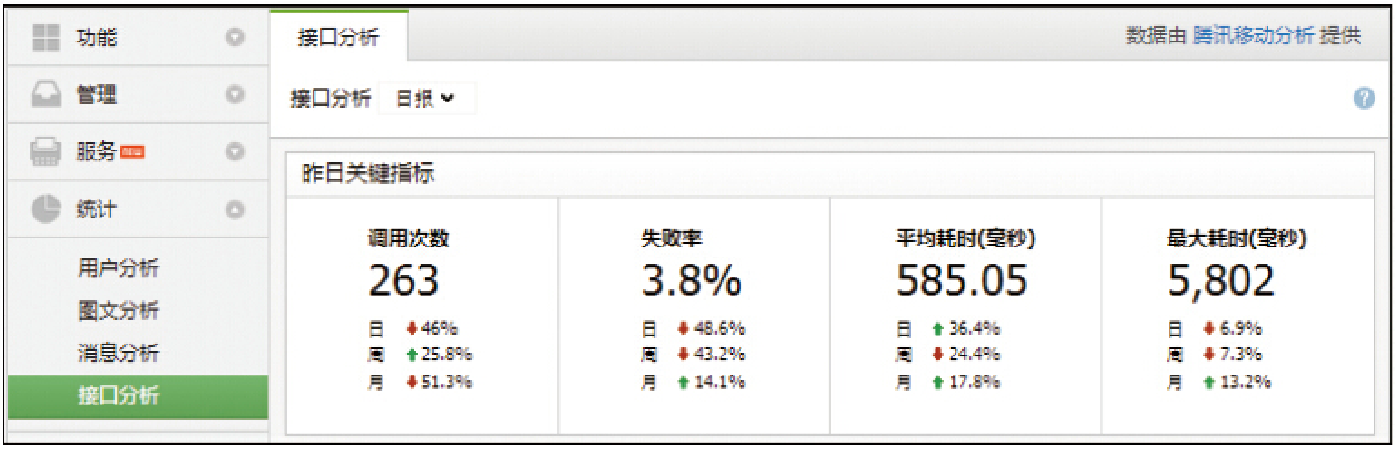


图1-33　接口分析

1.3.6　设置

1.账号信息

账号信息页面显示了公众账号的头像、名称、登录邮箱、原始ID、微信号、隐私设置、类型、认证情况、地区、功能介绍、腾讯微博、二维码、图片水印等信息。图1-34显示了账号的部分信息。

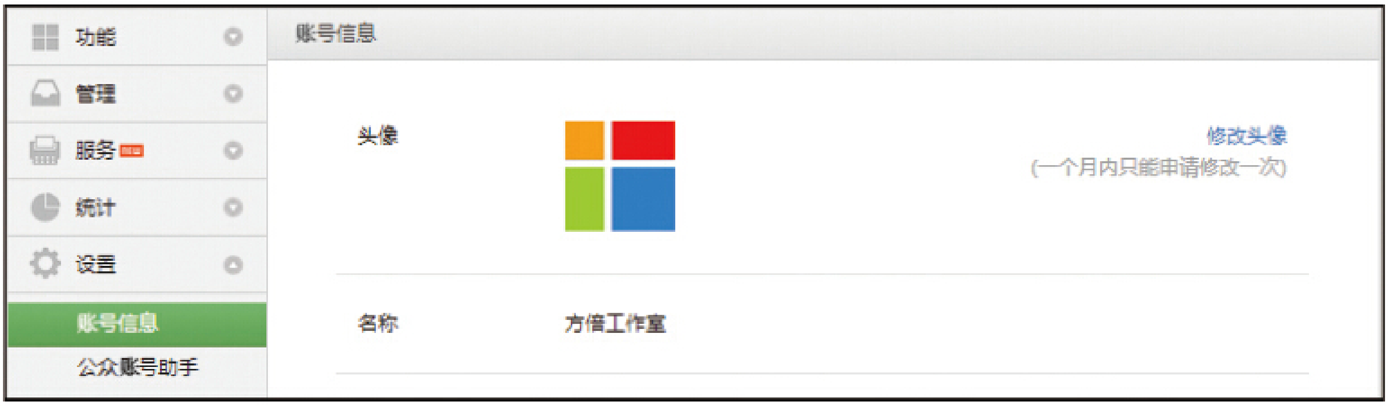


图1-34　账号信息

二维码是用户关注公众账号的一个重要入口，也是企业在微信公众平台上对外进行传播推广时的一张重要名片。微信公众平台提供五种不同尺寸的二维码供运营者下载。方倍工作室的二维码图片如图1-35所示。读者可以使用微信的“扫一扫”功能，扫描二维码关注我们。

图1-35　二维码

2.微信认证

通过微信认证的账号，可以看到微信认证的日期及已获得的权限列表，如图1-36所示。

图1-36　微信认证

3.公众账号助手

公众账号助手可以用来代替公众平台群发消息，并可随时查看消息群发状态。运营者可以用个人微信号来确认绑定，绑定后个人微信号将自动关注公众账号安全助手。以后向公众账号安全助手发送的内容，系统将自动群发给所有关注公众账号的用户。绑定界面如图1-37所示。

图1-37　公众账号助手

4.安全中心

安全中心可以开启手机保护功能，开通手机保护后，登录时需要输入手机验证码进行验证后才可正常登录。启用手机保护设置如图1-38所示。

图1-38　安全中心

1.3.7　开发者中心

开发者中心提供了开发者ID，其中包括AppID（应用ID）和AppSecret（应用密钥）。这些账号用于高级接口及微信支付的开发。同时也可以配置服务器的URL（服务器地址）、Token（令牌）、EncodingAESKey（消息加解密密钥）及消息加解密方式。具体后台界面如图1-39所示。

图1-39　开发者中心

1.4 本章小结

本章概要介绍了微信及其相关的三大平台，重点且详细介绍了当今最流行、最热门的微信公众账号的注册及微信公众平台的使用方法，开发者及运营者应该掌握这些基本功能并熟悉它们的使用方法，以便为后续的开发运营打下基础。

第2章 微信开发者中心

启用微信公众平台开发者中心需要拥有自己的服务器资源，用于存放自己开发的程序文件。服务器可以是一个虚拟空间，也可以是个云主机或云空间，只要这个空间支持程序的运行并且有域名或者拥有可有效访问的IP。在完成程序开发后，开发者需要把程序上传到服务器，这样才能被微信访问到。

本章以新浪云SAE为程序运行环境，介绍如何申请开通自己的SAE应用以及如何将SAE和微信公众账号进行对接。

2.1 新浪SAE应用

Sina App Engine（以下简称SAE）是新浪研发中心于2009年8月开始内部开发，并在2009年11月3日正式推出第一个Alpha版本的国内首个公有云计算平台（http://sae.sina.com.cn），SAE是新浪云计算战略的核心组成部分。

SAE作为国内的公有云计算，从开发伊始借鉴吸纳Google、Amazon等国外公司的公有云计算的成功技术经验，并很快推出不同于他们的具有自身特色的云计算平台。SAE选择在国内流行最广的Web开发语言PHP作为首选支持语言，Web开发者可以在Linux/Mac/Windows上通过SVN或者Web版在线代码编辑器进行开发、部署、调试，团队开发时还可以进行成员协作，不同的角色将对代码、项目拥有不同的权限；SAE提供了一系列分布式计算、存储服务供开发者使用，包括分布式文件存储、分布式数据库集群、分布式缓存、分布式定时服务等，这些服务将大大降低开发者的开发成本。同时SAE整体架构的高可靠性和新浪的品牌保证大大降低了开发者的运营风险。另外，作为典型的云计算，SAE采用“所付即所用，所付仅所用”的计费理念，通过日志和统计中心精确地计算每个应用的资源消耗（包括CPU、内存、磁盘等）。

总之，SAE就是简单高效的分布式Web服务开发、运行平台。

2.1.1 申请新浪云账号

SAE的注册地址为：http://sae.sina.com.cn/?m=user&a=reg，在浏览器中输入该网址，将自动跳转到“SAE新浪云计算平台”在新浪微博的应用授权界面，如图2-1所示。

图2-1 SAE登录界面

如果你还没有新浪微博账号，需要新注册一个，注册新浪微博账号的过程本书就不再介绍了。在图2-1中填写新浪微博账号及密码之后，单击“登录”按钮，将跳转到授权确认页面，如图2-2所示。

图2-2　SAE新浪云计算平台授权确认页面

单击“授权”按钮，跳转到SAE的用户注册页面，如图2-3所示。

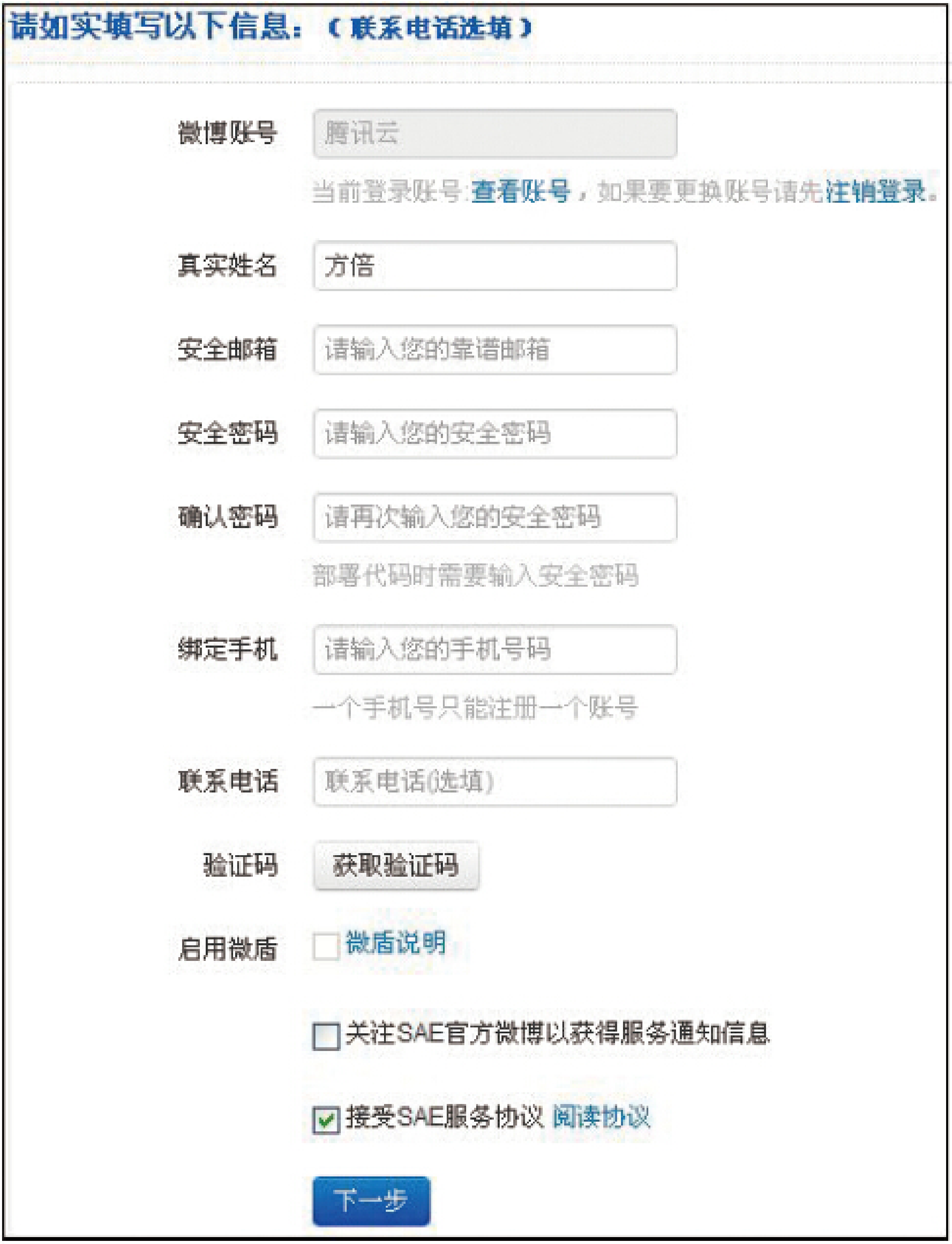


图2-3　SAE用户注册页面

填写真实姓名、安全邮箱、安全密码、确认密码、绑定手机及验证码之后，单击“下一步”按钮，将跳转到手机号验证页面，如图2-4所示。

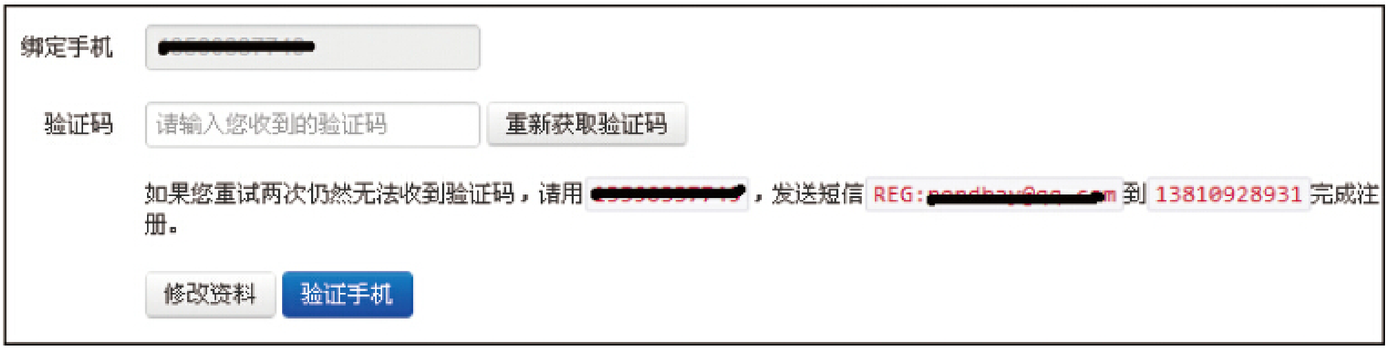


图2-4　SAE手机认证页面

填写手机收到的短信验证码后，单击“验证手机”按钮，将提示注册成功。这样，我们成功注册了SAE的账号。

2.1.2　创建新浪云应用

使用注册成功的微博账号登录SAE，登录后界面如图2-5所示。

图2-5　SAE首页页面

如图2-6所示，在最上方右侧的导航列表中，单击“我的应用”链接，再从下拉列表中选择“应用列表”链接，将跳转到“应用列表”页面。

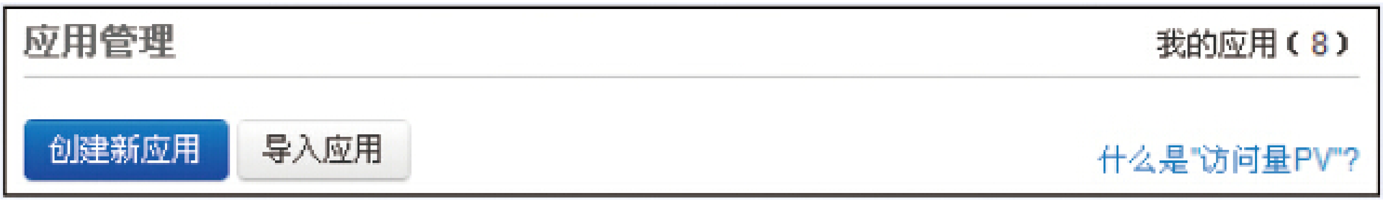


图2-6　应用列表

单击页面中的“创建新应用”按钮，这时会弹出提示框，提示禁止放置违法违规内容，单击“继续创建”按钮，弹出创建应用页面，如图2-7所示。

图2-7　创建应用

在应用创建页面中，依次填写二级域名（App name）、应用名称、验证码，开发语言选择“PHP”，应用类型单击“PHP空应用”。如果二级域名（App name）已经被其他人注册过，会提示已经被占用，需要重新选择填入。填写完毕后，单击“创建应用”按钮，将提示创建成功，如图2-8所示。

图2-8 应用创建成功

应用创建成功之后，会自动跳转到应用列表，在应用列表中可以看到已经创建的应用。

2.1.3 创建应用版本

在应用列表中，单击刚才创建的应用cctv15将进入应用信息页面，如图2-9所示。

图2-9 应用信息

如图2-10所示，找到左侧目录下的“应用管理”，再单击下面的“代码管理”链接，将跳转到代码管理页面。

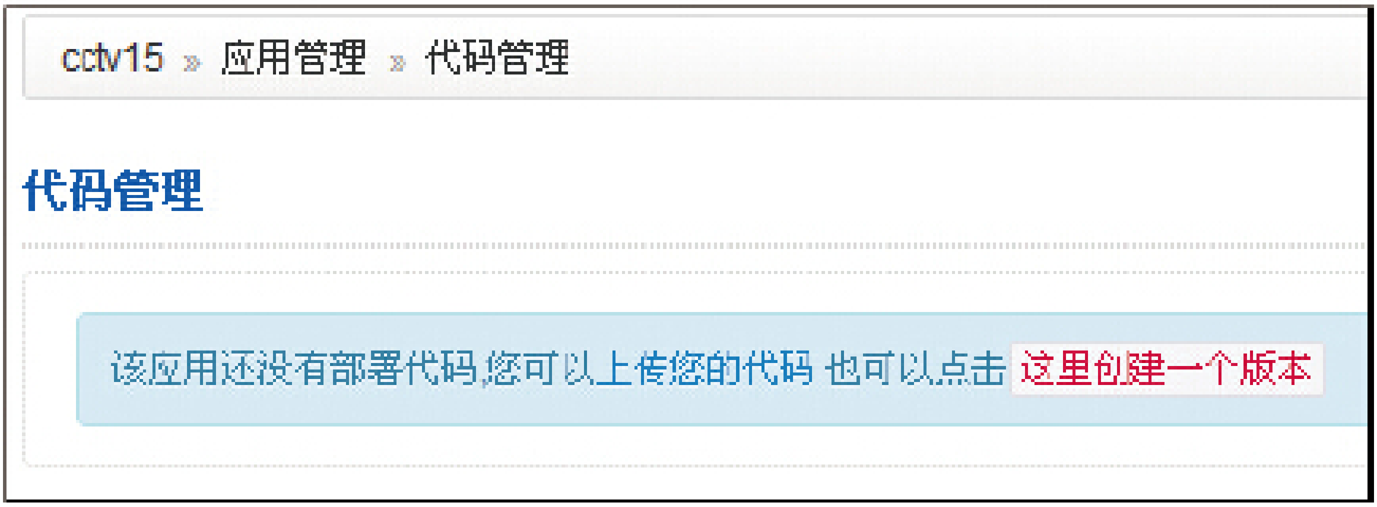


图2-10　代码管理

单击右侧的“创建一个版本”按钮，将弹出创建版本页面，如图2-11所示。

图2-11　创建版本

版本号默认为1，可以不用更改，直接单击“创建”按钮，这时会弹出安全密码输入框，正确输入安全密码之后，如图2-12所示。

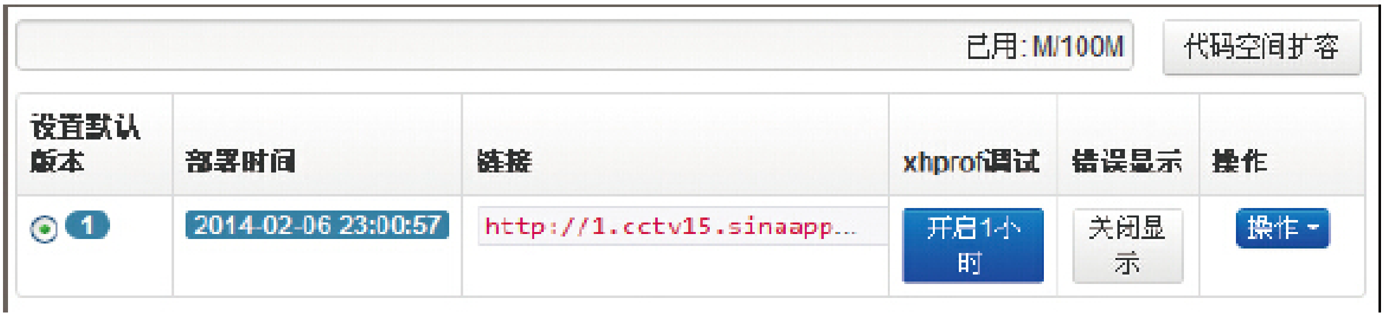


图2-12　创建版本成功

至此，就成功创建了一个域名URL为http://cctv15.sinaapp.com/的SAE应用了。这个URL将会在后面用到。

2.1.4　上传微信接口代码

下述代码是一个可以启用微信接口的代码。

```
<?php
/*
    方倍工作室 http://www.fangbei.org/
    CopyRight 2013 www.doucube.com  All Rights Reserved
*/
define("TOKEN", "weixin");
$wechatObj = new wechatCallbackapiTest();
if (isset($_GET['echostr'])) {
    $wechatObj->valid();
```

```
}else{
    $wechatObj->responseMsg();
}
class wechatCallbackapiTest
{
    public function valid()
    {
        $echoStr = $_GET["echostr"];
        if($this->checkSignature()){
            echo $echoStr;
            exit;
        }
    }
    private function checkSignature()
    {
        $signature = $_GET["signature"];
        $timestamp = $_GET["timestamp"];
        $nonce = $_GET["nonce"];
        $token = TOKEN;
        $tmpArr = array($token, $timestamp, $nonce);
        sort($tmpArr);
        $tmpStr = implode( $tmpArr );
        $tmpStr = sha1( $tmpStr );
        if( $tmpStr == $signature ){
            return true;
        }else{
            return false;
        }
    }
    public function responseMsg()
    {
        $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
        if (!empty($postStr)){
            $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
            $fromUsername = $postObj->FromUserName;
            $toUsername = $postObj->ToUserName;
            $keyword = trim($postObj->Content);
            $time = time();
            $textTpl = "<xml>
                        <ToUserName><![CDATA[%s]]></ToUserName>
                        <FromUserName><![CDATA[%s]]></FromUserName>
                        <CreateTime>%s</CreateTime>
                        <MsgType><![CDATA[%s]]></MsgType>
                        <Content><![CDATA[%s]]></Content>
                        <FuncFlag>0</FuncFlag>
                        </xml>";
            if($keyword == "?" || $keyword == "? ")
            {
                $msgType = "text";
                $content = date("Y-m-d H:i:s",time());
                $result = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, $content);
                echo $result;
            }
        }else{
            echo "";
            exit;
        }
    }
}
?>
```

将代码保存到文件index.php。请注意必须使用专业的开发软件来保存操作，例如Notepad++，不要使用Windows自带的记事本等，保存时需设置格式为“UTF-8无BOM格式”。图2-13是Notepad++的设置处。

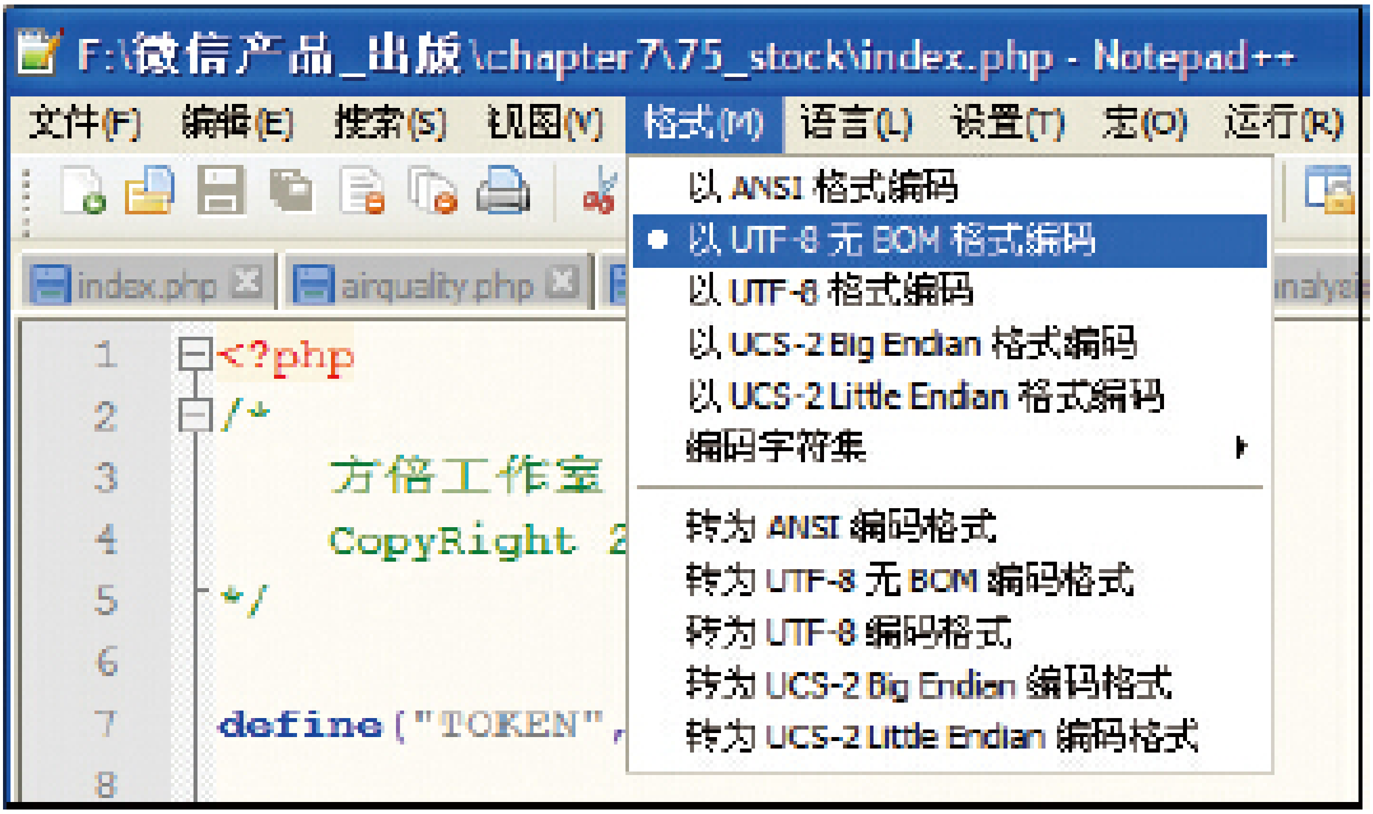


图2-13　UTF-8无BOM格式编码

然后使用压缩软件WinRAR将其压缩成ZIP格式，注意不能用RAR格式，因为SAE不支持RAR格式的文件上传，如图2-14所示。

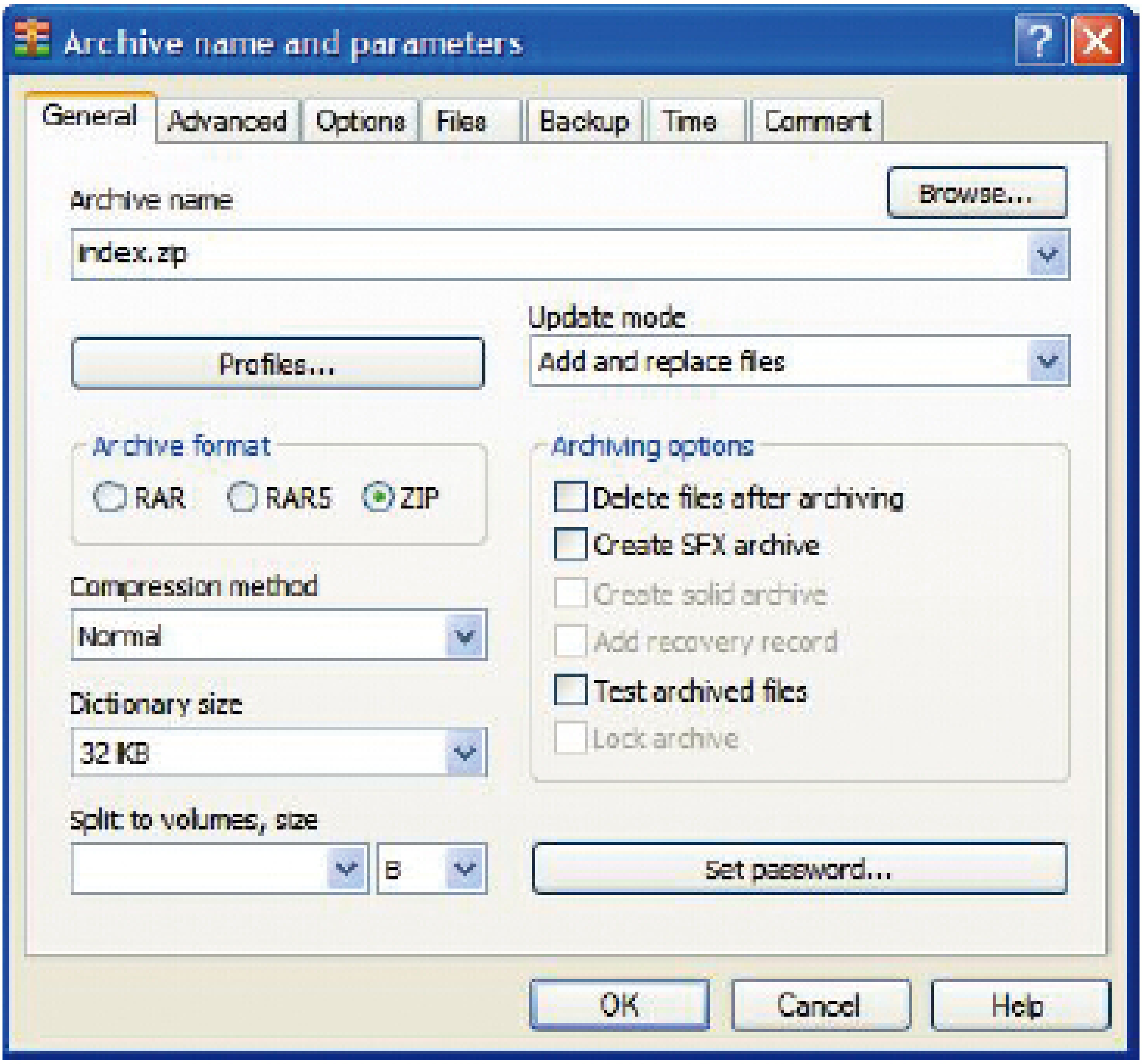


图2-14　压缩zip文件

这样就会生成一个index.zip的压缩文件。

再回到我们之前创建的SAE应用的代码管理界面中，单击“操作”按钮，选择在下拉按钮中，再选择“上传代码包”按钮，如图2-15所示。

图2-15　选择上传代码包

选择上传代码包后，将弹出“代码上传”页面，单击“上传文件”按钮，选择刚才压缩好的index.zip文件，单击上传，上传成功后进度条的背景色为绿色，如图2-16所示。

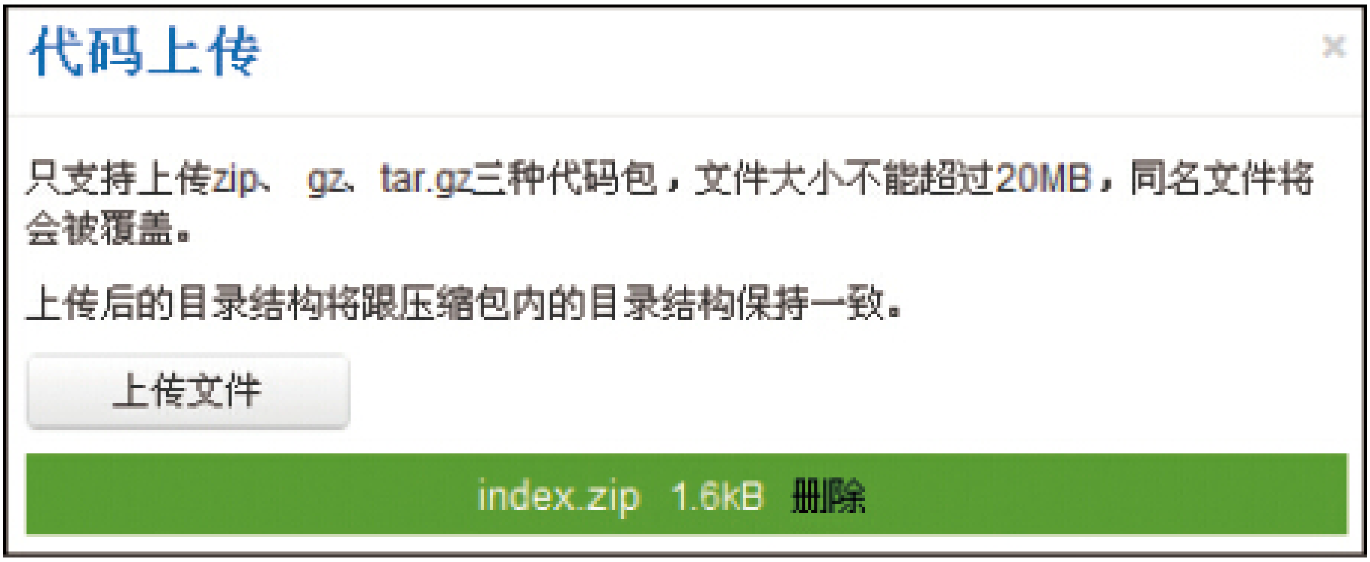


图2-16　上传代码

再次回到代码管理界面中，单击“编辑代码”按钮，如图2-17所示。

图2-17　编辑代码

在新的代码编辑页面中，我们可以看到index.php已经上传成功，双击该文件可以查看到源代码，如图2-18所示。

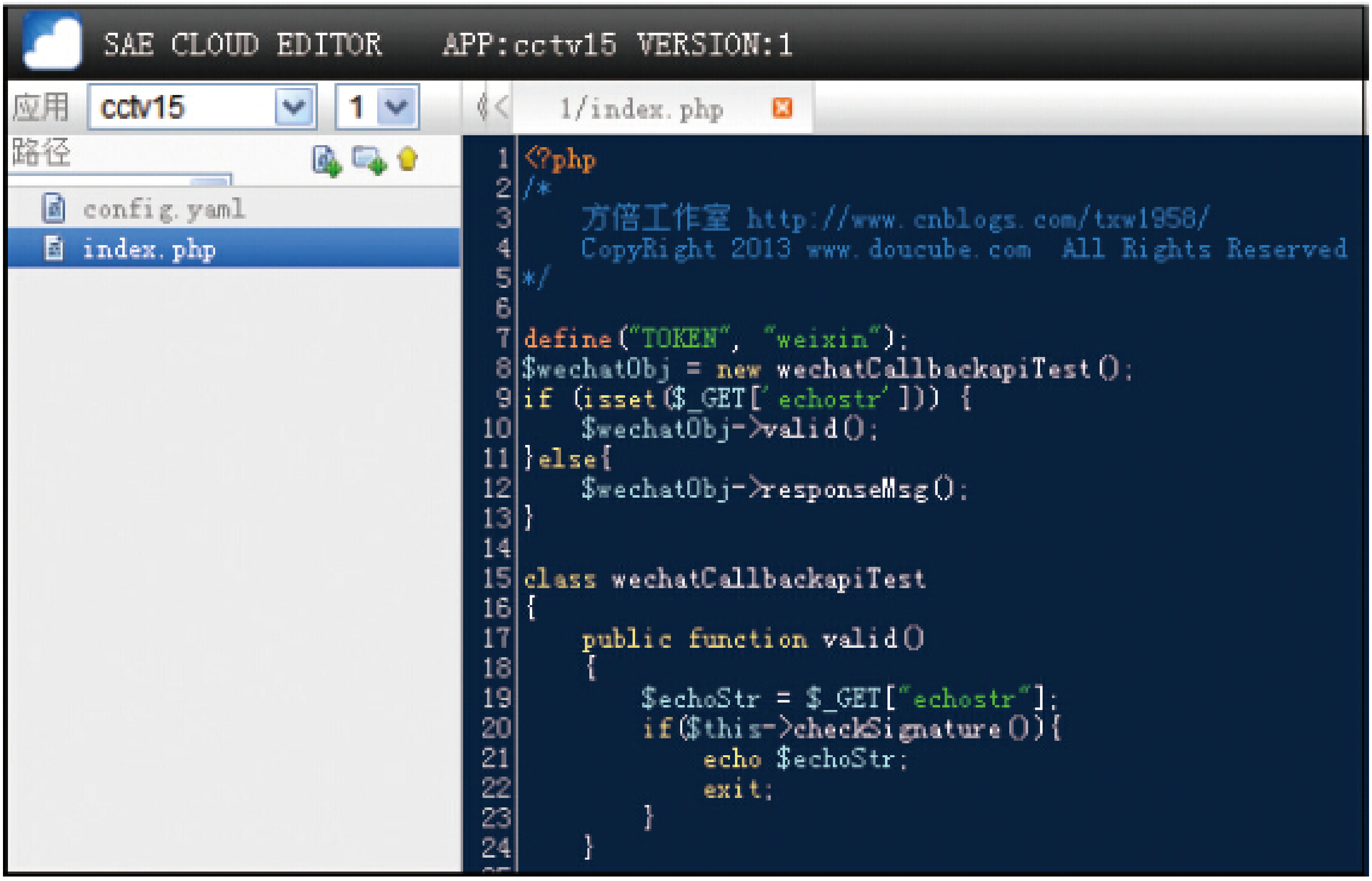


图2-18　查看源代码

至此，新浪云应用的创建完成，并且成功上传了微信公众平台的接口文件。

2.1.5 申请SAE中级认证

为了更好地利用云计算的资源，帮助更多开发者减轻服务器成本负担及推广微信公众平台的开发，新浪云SAE运营中心和方倍工作室进行合作，凡是购买了本书纸质版的读者（购买地址包括方倍工作室博客首页、各地新华书店及天猫、京东、当当、亚马逊等网上商城）在成功开发出一种微信公众平台的功能之后，可以申请新浪SAE中级认证，而无须达到SAE官网上要求的认证条件。在SAE审核通过之后即可获得新浪SAE的中级认证，每月可获得10000云豆用于微信公众平台的运营（价值1200元/年）。所以，读者可以在学习完本书并开发出一个微信应用之后申请认证。

申请表可以从本书的介绍页面上找到，地址为：

http://www.cnblogs.com/txw1958/p/weixin-development-best-practice.html

2.2 微信开发者中心

2.2.1 配置和启用服务器

登录微信公众平台后台，微信公众平台地址：http://mp.weixin.qq.com，在左侧列表中最下方，找到“开发者中心”，如图2-19所示。

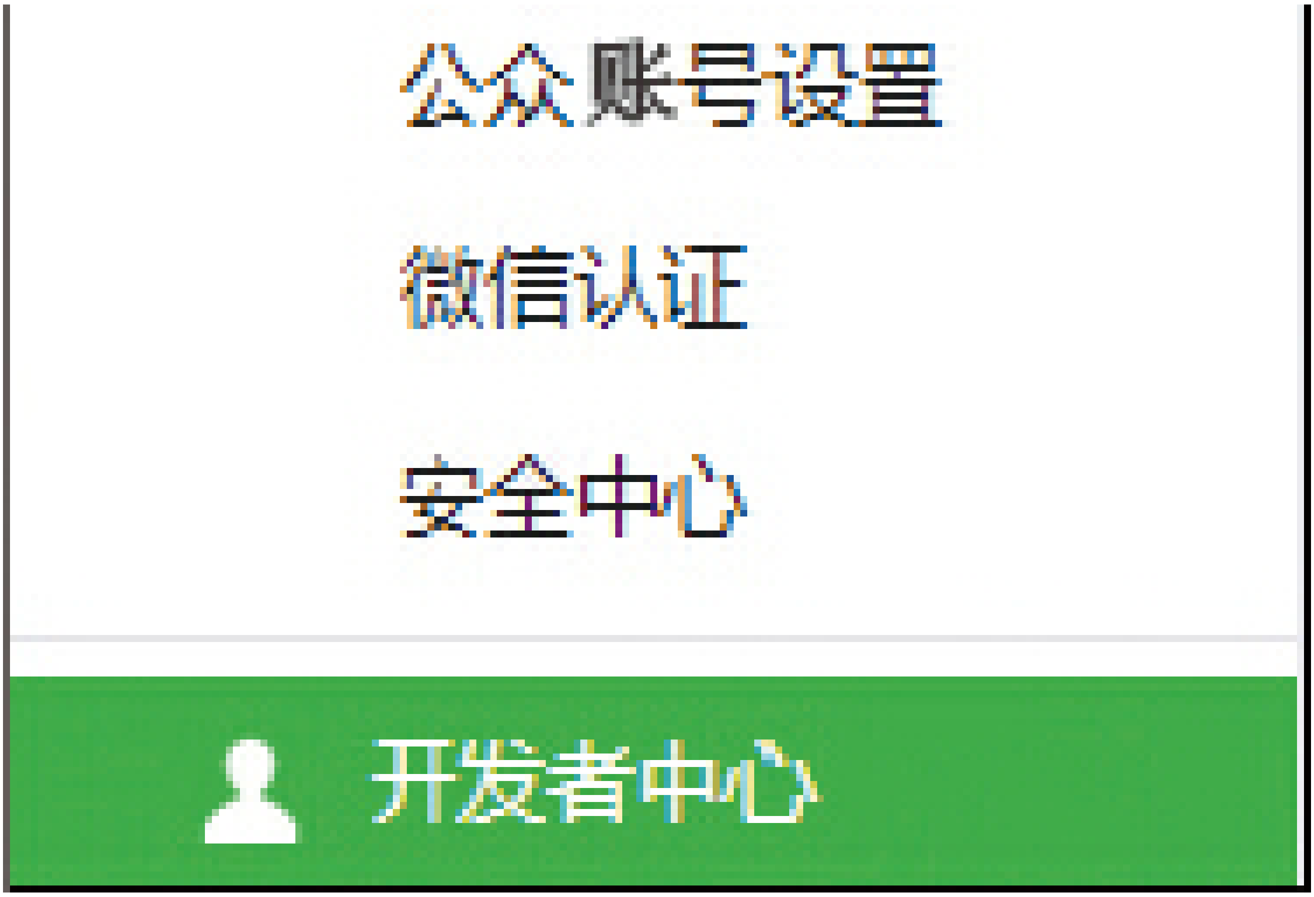


图2-19　开发者中心

单击进入开发者中心，可以看到当前有个服务器配置的信息，状态为未启用，如图2-20所示。

图2-20　未启用服务器配置

单击“修改配置”按钮，进入配置页面，如图2-21所示。

图2-21　填写服务器配置

此处的URL为上一节中介绍的云应用的域名，即为http://cctv15.sinaapp.com，而Token在index.php中定义为weixin，EncodingAESKey不需要填写，单击“随机生成”按钮，让系统自动生成一个即可，消息加解密方式选择“明文模式”，然后单击“提交”按钮。弹出确认框，如图2-22所示。

图2-22　确定提交配置

在弹出的提示框中，单击“确定”按钮，相关参数填写成功，如图2-23所示。

图2-23　服务器已配置

再单击右上角的“启用”按钮来启用服务器的配置。系统弹出提示框，询问是否确定开启服务器配置，如图2-24所示。

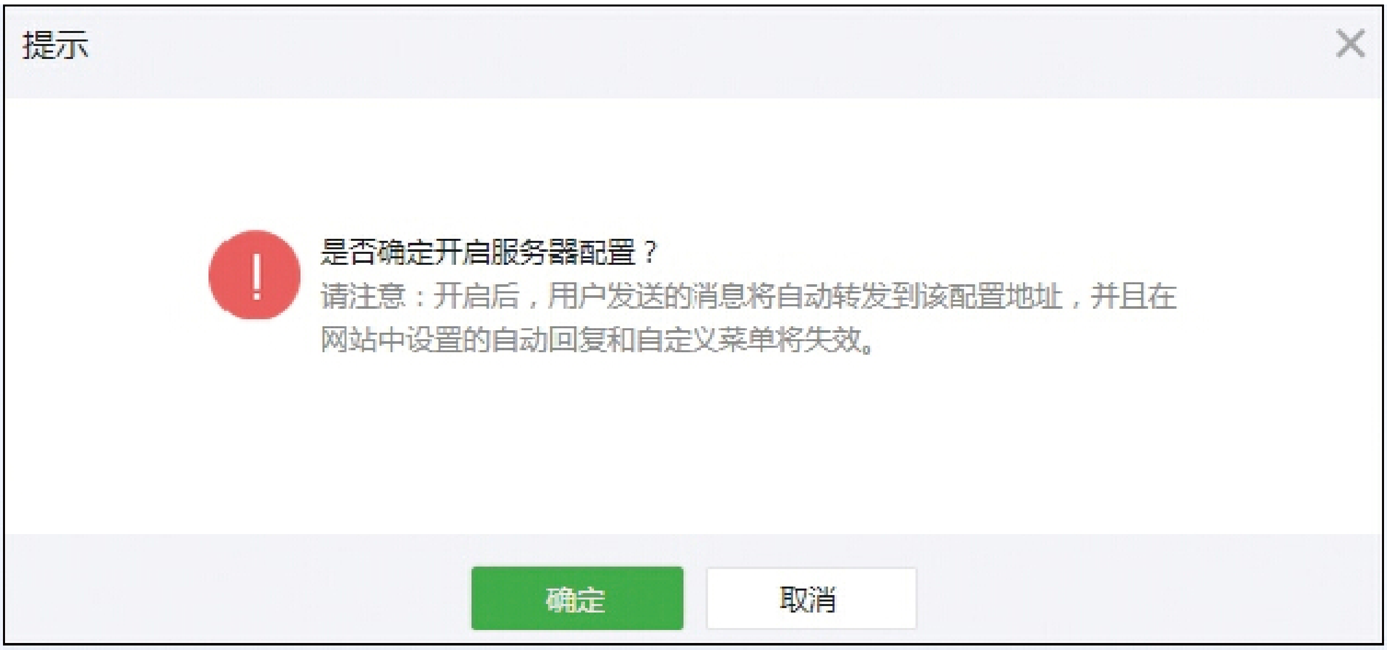


图2-24　确定开启服务器配置

单击“确定”按钮将启用服务器配置。

如果单击按钮后，上方提示“token验证失败”，可以重试几次，有时候微信服务器不稳定也会造成这样的情况，并不是程序本身有问题。启用成功后界面如图2-25所示。

图2-25　服务器配置已启用

这样就成功配置并启用了服务器。

2.2　微信开发者中心

2.2.1　配置和启用服务器

登录微信公众平台后台，微信公众平台地址：http://mp.weixin.qq.com，在左侧列表中最下方，找到“开发者中心”，如图2-19所示。

图2-19　开发者中心

单击进入开发者中心，可以看到当前有个服务器配置的信息，状态为未启用，如图2-20所示。

图2-20　未启用服务器配置

单击“修改配置”按钮，进入配置页面，如图2-21所示。

图2-21　填写服务器配置

此处的URL为上一节中介绍的云应用的域名，即为http://cctv15.sinaapp.com，而Token在index.php中定义为weixin，EncodingAESKey不需要填写，单击“随机生成”按钮，让系统自动生成一个即可，消息加解密方式选择“明文模式”，然后单击“提交”按钮。弹出确认框，如图2-22所示。

图2-22　确定提交配置

在弹出的提示框中，单击“确定”按钮，相关参数填写成功，如图2-23所示。

图2-23　服务器已配置

再单击右上角的“启用”按钮来启用服务器的配置。系统弹出提示框，询问是否确定开启服务器配置，如图2-24所示。

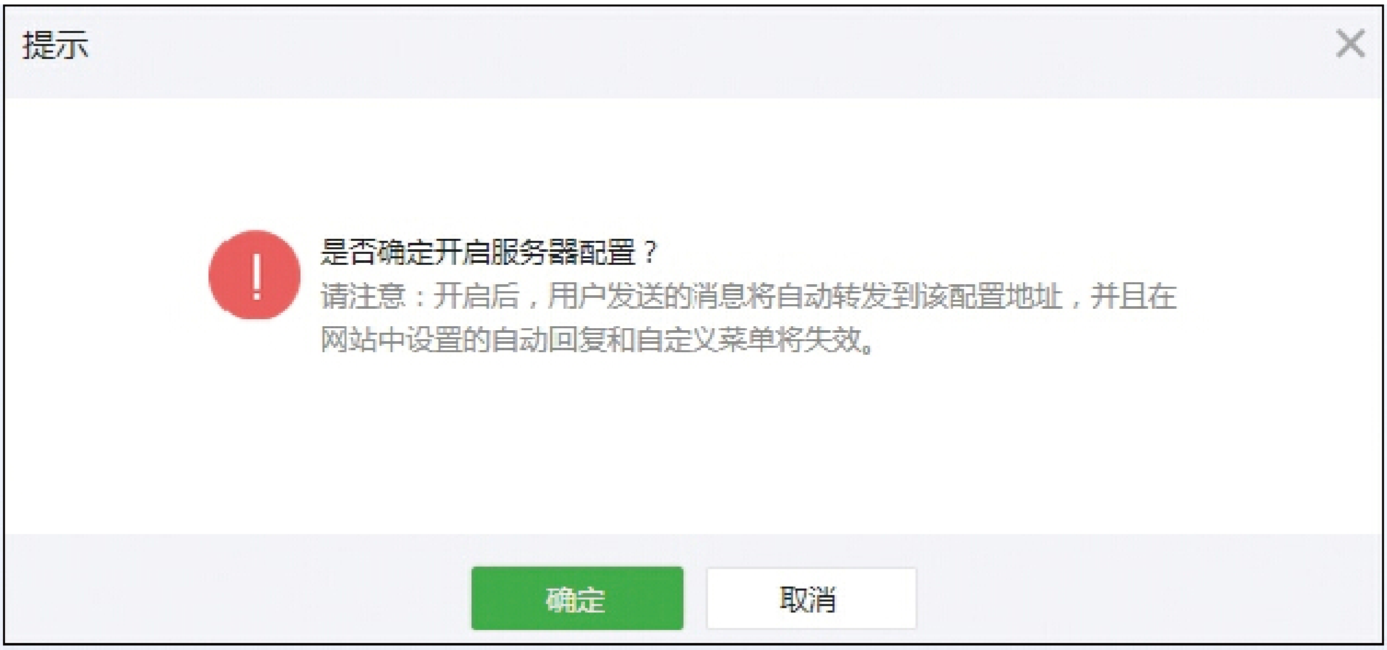


图2-24　确定开启服务器配置

单击“确定”按钮将启用服务器配置。

如果单击按钮后，上方提示“token验证失败”，可以重试几次，有时候微信服务器不稳定也会造成这样的情况，并不是程序本身有问题。启用成功后界面如图2-25所示。

图2-25　服务器配置已启用

这样就成功配置并启用了服务器。

2.2.2　消息体加解密实现

在图2-21中，微信公众平台在配置服务器时，提供了3种加解密的模式供开发者选择，即明文模式、兼容模式、安全模式，选择兼容模式和安全模式前，需在开发者中心填写AES对称加密算法的消息加解密密钥EncodingAESKey。公众账号用此密钥对收到的密文消息体进行解密，回复消息体也用此密钥加密。

· 明文模式：维持现有模式，没有适配加解密新特性，消息体明文收发，默认设置为明文模式。

· 兼容模式：公众平台发送消息内容将同时包括明文和密文，消息包长度增加到原来的3倍左右；公众账号回复明文或密文均可，不影响现有消息收发；开发者可在此模式下进行调试。

· 安全模式（推荐）：公众平台发送消息体的内容只含有密文，公众账号回复的消息体也为密文，建议开发者在调试成功后使用此模式收发消息。

消息体加解密的实现过程如下：

假设本次的开发配置中URL为

```
http://www.fangbei.org/index.php
```

接口程序中需要配置以下三项参数：

```
/*
    方倍工作室 http://www.cnblogs.com/txw1958/
    CopyRight 2014 All Rights Reserved
*/
define("TOKEN", "weixin");
define("AppID", "wxbad0b45542aa0b5e");
define("EncodingAESKey", "abcdefghijklmnopqrstuvwxyz0123456789ABCDEFG");
require_once('wxBizMsgCrypt.php');
```

当用户向公众账号发送消息时，微信公众账号将会在URL中带上signature、timestamp、nonce、encrypt_type、msg_signature等参数，如下所示：

```
http://www.fangbei.org/index.php?signature=35703636de2f9df2a77a662b68e521ce17c34db4&timestamp=1414243737&nonce=1792106704&encrypt_type=aes&msg_signature=6147984331daf7a1a9eed6e
```

同时向该接口推送如下XML消息，即一个已加密的消息。

```
<xml>
  <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
  <Encrypt><![CDATA[MNn4+jJ/VsFh2gUyKAaOJArwEVYCvVmyN0iXzNarP3O6vXzK62ft1/KG2/XPZ4y5bPWU/jfIfQxODRQ7sLkUsrDRqsWimuhIT8Eq+w4E/28m+XDAQKEOjWTQIOp1p6kNsIV1DdC3B+AtcKcKSNAeJDr7x7GH
</xml>
```

这时，程序需要从URL中获得以下参数：

```
$timestamp  = $_GET['timestamp'];
$nonce = $_GET["nonce"];
$msg_signature  = $_GET['msg_signature'];
$encrypt_type = $_GET['encrypt_type'];
```

这些参数将用于加解密过程。

接口程序收到消息后，先进行解密，解密部分代码如下：

```
$postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
if ($encrypt_type == 'aes'){
    $pc = new WXBizMsgCrypt(TOKEN, EncodingAESKey, AppID);
    $this->logger(" D \r\n".$postStr);
    $decryptMsg = "";  //解密后的明文
    $errCode = $pc->DecryptMsg($msg_signature, $timestamp, $nonce, $postStr, $decryptMsg);
    $postStr = $decryptMsg;
}
```

解密完成后，把解密内容又返回给$postStr，这是为了保证将消息中解密后的内容和明文模式时的消息统一，方便后续处理，解密后的XML如下：

```
<xml>
    <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
    <FromUserName><![CDATA[oIDrpjpQ8j8mBuQ8nM26HWzNEZgg]]></FromUserName>
    <CreateTime>1414243737</CreateTime>
```

```
    <MsgType><![CDATA[text]]></MsgType>
    <Content><![CDATA[?]]></Content>
    <MsgId>6074130599188426998</MsgId>
</xml>
```

在原有的代码流程中对消息进行处理，完成之后，一个要回复的文本消息如下：

```
<xml>
    <ToUserName><![CDATA[oIDrpjpQ8j8mBuQ8nM26HWzNEZgg]]></ToUserName>
    <FromUserName><![CDATA[gh_680bdefc8c5d]]></FromUserName>
    <CreateTime>1414243733</CreateTime>
    <MsgType><![CDATA[text]]></MsgType>
    <Content><![CDATA[2014-10-25 21:28:53
技术支持 方倍工作室
http://www.fangbei.org/]]></Content>
</xml>
```

把上述消息进行加密，返回给微信公众账号，加密过程如下：

```
//加密
if ($encrypt_type == 'aes'){
    $encryptMsg = ''; //加密后的密文
    $errCode = $pc->encryptMsg($result, $timeStamp, $nonce, $encryptMsg);
    $result = $encryptMsg;
    $this->logger(" E \r\n".$result);
}
```

加密后的内容如下：

```
<xml>
<Encrypt><![CDATA[pE6gp6qvVBMHwCXwnM7illFBrh9LmvlKFlPUDuyQo9EKNunqbUFMd2KjiYoz+
3K1B+93JbMWHt+19TI8awdRdyopRS4oUNg5M2jwpwXTmc6TtafkKNjvqlvPXIWmutw0tuMXke1hDgsqz0SC8h/QjNLxECuwnczrfCMJlt+APHnX2yMMaq/aYUNcndOH387loQvl2suCGucXpglnbxf7frTCz9NQVgKiYrvKOhk6KFiVM
  <MsgSignature><![CDATA[6c46904dc1f58b2ddf2dd0399f1c6cf41f33ecb9]]></MsgSignature>
  <TimeStamp>1414243733</TimeStamp>
  <Nonce><![CDATA[1792106704]]></Nonce>
</xml>
```

这样，一个安全模式下的加解密消息就完成了。

完整的代码清单如下：

```
<?php
/*
    方倍工作室 http://www.cnblogs.com/txw1958/
    CopyRight 2014 All Rights Reserved
*/
define("TOKEN", "weixin");
define("AppID", "wxbad0b45542aa0b5e");
define("EncodingAESKey", "abcdefghijklmnopqrstuvwxyz0123456789ABCDEFG");
require_once('wxBizMsgCrypt.php');

$wechatObj = new wechatCallbackapiTest();
if (!isset($_GET['echostr'])) {
    $wechatObj->responseMsg();
}else{
    $wechatObj->valid();
}

class wechatCallbackapiTest
{
    //验证签名
    public function valid()
    {
        $echoStr = $_GET["echostr"];
        $signature = $_GET["signature"];
        $timestamp = $_GET["timestamp"];
        $nonce = $_GET["nonce"];
        $tmpArr = array(TOKEN, $timestamp, $nonce);
        sort($tmpArr);
        $tmpStr = implode($tmpArr);
        $tmpStr = sha1($tmpStr);
        if($tmpStr == $signature){
            echo $echoStr;
            exit;
        }
    }

    //响应消息
    public function responseMsg()
    {
        $timestamp  = $_GET['timestamp'];
        $nonce = $_GET["nonce"];
        $msg_signature  = $_GET['msg_signature'];
        $encrypt_type = (isset($_GET['encrypt_type']) && ($_GET['encrypt_type'] == 'aes')) ? "aes" : "raw";

        $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
        if (!empty($postStr)){
            //解密
            if ($encrypt_type == 'aes'){
                $pc = new WXBizMsgCrypt(TOKEN, EncodingAESKey, AppID);
                $this->logger(" D \r\n".$postStr);
                $decryptMsg = "";  //解密后的明文
                $errCode = $pc->DecryptMsg($msg_signature, $timestamp, $nonce, $postStr, $decryptMsg);
                $postStr = $decryptMsg;
            }
            $this->logger(" R \r\n".$postStr);
            $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
            $RX_TYPE = trim($postObj->MsgType);

            //消息类型分离
            switch ($RX_TYPE)
            {
                case "event":
                    $result = $this->receiveEvent($postObj);
                    break;
                case "text":
                    $result = $this->receiveText($postObj);
                    break;
            }
            $this->logger(" R \r\n".$result);
            //加密
            if ($encrypt_type == 'aes'){
                $encryptMsg = ''; //加密后的密文
                $errCode = $pc->encryptMsg($result, $timeStamp, $nonce, $encryptMsg);
                $result = $encryptMsg;
                $this->logger(" E \r\n".$result);
            }
            echo $result;
        }else {
            echo "";
            exit;
        }
    }

```

```php
    //接收事件消息
    private function receiveEvent($object)
    {
        $content = "";
        switch ($object->Event)
        {
            case "subscribe":
                $content = "欢迎关注方倍工作室 ";
                break;
        }

        $result = $this->transmitText($object, $content);
        return $result;
    }

    //接收文本消息
    private function receiveText($object)
    {
        $keyword = trim($object->Content);
        if (strstr($keyword, "文本")){
            $content = "这是个文本消息";
        }else if (strstr($keyword, "单图文")){
            $content = array();
            $content[] = array("Title"=>"单图文标题",  "Description"=>"单图文内容", "PicUrl"=>"http://discuz.comli.com/weixin/weather/icon/cartoon.jpg", "Url" =>"http://m.cnb
        }else if (strstr($keyword, "图文") || strstr($keyword, "多图文")){
            $content = array();
            $content[] = array("Title"=>"多图文1标题", "Description"=>"", "PicUrl" =>"http://discuz.comli.com/weixin/weather/icon/cartoon.jpg", "Url" =>"http://m.cnblogs.com
            $content[] = array("Title"=>"多图文2标题", "Description"=>"", "PicUrl" =>" http://d.hiphotos.bdimg.com/wisegame/pic/item/f3529822720e0cf3ac9f1ada0846f21fbe09aaa3
            $content[] = array("Title"=>"多图文3标题", "Description"=>"", "PicUrl"=>"http://g.hiphotos.bdimg.com/wisegame/pic/item/18cb0a46f21fbe090d338acc6a600c338644adfd.j
        }else if (strstr($keyword, "音乐")){
            $content = array();
            $content = array("Title"=>"最炫民族风", "Description"=>"歌手：凤凰传奇", "MusicUrl"=>"http://121.199.4.61/music/zxmzf.mp3", "HQMusicUrl"=>"http://121.199.4.61/musi
        }else{
            $content = date("Y-m-d H:i:s",time())."\n".$object->FromUserName."\n技术支持 方倍工作室";
        }

        if(is_array($content)){
            if (isset($content[0])){
                $result = $this->transmitNews($object, $content);
            }else if (isset($content['MusicUrl'])){
                $result = $this->transmitMusic($object, $content);
            }
        }else{
            $result = $this->transmitText($object, $content);
        }
        return $result;
    }

    //回复文本消息
    private function transmitText($object, $content)
    {
        $xmlTpl = "<xml>
    <ToUserName><![CDATA[%s]]></ToUserName>
    <FromUserName><![CDATA[%s]]></FromUserName>
    <CreateTime>%s</CreateTime>
    <MsgType><![CDATA[text]]></MsgType>
    <Content><![CDATA[%s]]></Content>
</xml>";
        $result = sprintf($xmlTpl, $object->FromUserName, $object->ToUserName, time(), $content);
        return $result;
    }

    //回复图文消息
    private function transmitNews($object, $newsArray)
    {
        if(!is_array($newsArray)){
            return;
        }
        $itemTpl = "        <item>
            <Title><![CDATA[%s]]></Title>
            <Description><![CDATA[%s]]></Description>
            <PicUrl><![CDATA[%s]]></PicUrl>
            <Url><![CDATA[%s]]></Url>
        </item>
";
        $item_str = "";
        foreach ($newsArray as $item){
            $item_str .= sprintf($itemTpl, $item['Title'], $item['Description'], $item['PicUrl'], $item['Url']);
        }
        $xmlTpl = "<xml>
    <ToUserName><![CDATA[%s]]></ToUserName>
    <FromUserName><![CDATA[%s]]></FromUserName>
    <CreateTime>%s</CreateTime>
    <MsgType><![CDATA[news]]></MsgType>
    <ArticleCount>%s</ArticleCount>
    <Articles>
$item_str    </Articles>
</xml>";

        $result = sprintf($xmlTpl, $object->FromUserName, $object->ToUserName, time(), count($newsArray));
        return $result;
    }

    //回复音乐消息
    private function transmitMusic($object, $musicArray)
    {
        $itemTpl = "<Music>
        <Title><![CDATA[%s]]></Title>
        <Description><![CDATA[%s]]></Description>
        <MusicUrl><![CDATA[%s]]></MusicUrl>
        <HQMusicUrl><![CDATA[%s]]></HQMusicUrl>
    </Music>";

        $item_str = sprintf($itemTpl, $musicArray['Title'], $musicArray['Description'], $musicArray['MusicUrl'], $musicArray['HQMusicUrl']);

        $xmlTpl = "<xml>
    <ToUserName><![CDATA[%s]]></ToUserName>
    <FromUserName><![CDATA[%s]]></FromUserName>
    <CreateTime>%s</CreateTime>
    <MsgType><![CDATA[music]]></MsgType>
    $item_str
</xml>";

        $result = sprintf($xmlTpl, $object->FromUserName, $object->ToUserName, time());
        return $result;
    }

    //日志记录
    public function logger($log_content)
    {
        if(isset($_SERVER['HTTP_APPNAME'])){   //SAE
            sae_set_display_errors(false);
            sae_debug($log_content);
            sae_set_display_errors(true);
        }else if($_SERVER['REMOTE_ADDR'] != "127.0.0.1"){ //LOCAL
            $max_size = 500000;
            $log_filename = "log.xml";
            if(file_exists($log_filename) and (abs(filesize($log_filename)) > $max_size)){unlink($log_filename);}
            file_put_contents($log_filename, date('Y-m-d H:i:s').$log_content."\r\n", FILE_APPEND);
        }
    }
}
?>
```

2.2.3 配置失败常见问题与分析

当我们在提交URL和Token的时候，有时候会碰到提交不成功的情况，具体有以下几种：

1）请求URL超时。

这种情况一般是由于服务器网速或响应速度太慢。可以先重试几次或者等一段时间再来试，如果还是这样，则需要考虑更换速度更快、性能更好的服务器。

2）系统发生错误，请稍后重试。

这种情况一般是由于微信服务器短时间内异常引起的，重试或者过一段时间尝试即可。

3）Token验证失败。

这种情况需要具体分析验证过程被卡在哪一个环节了。我们可以通过调用变量$_SERVER来获取服务器和执行环境信息来分析。

$_SERVER是一个包含了诸如头信息（header）、路径（path），以及脚本位置（script locations）等信息的数组。这个数组中的项目由Web服务器创建。了解更多关于$_SERVER的信息，可访问官方网站：

http://www.php.net/manual/zh/reserved.variables.server.php

这里我们需要使用以下两个元素：

· $_SERVER[‘REMOTE_ADDR’]（来访者的IP地址，此处为微信服务器的IP）

· $_SERVER[‘QUERY_STRING’]（查询请求字符串，此处为微信服务器发过来的GET请求字符串）

将以上两个变量记录到日志中。函数定义如下：

```
function traceHttp()
{
    $content = date('Y-m-d H:i:s')."\nREMOTE_ADDR:".$_SERVER["REMOTE_ADDR"]."\nQUERY_STRING:".$_SERVER["QUERY_STRING"]."\n\n";
    if (isset($_SERVER['HTTP_APPNAME'])){   //SAE
        sae_set_display_errors(false);
        sae_debug(trim($content));
        sae_set_display_errors(true);
    }else {
        $max_size = 100000;
        $log_filename = "log.xml";
        if(file_exists($log_filename) and (abs(filesize($log_filename)) > $max_size)){unlink($log_filename);}
        file_put_contents($log_filename, $content, FILE_APPEND);
    }
}
```

上面代码中，当环境为SAE时，使用SAE的调试函数sae_debug（）将内容记录到日志中。而在具有读写权限的空间下，使用file_put_contents（）函数把字符串写入文件。

然后在程序的数据处理之前调用该函数，记录信息，代码如下所示：

```
define("TOKEN", "weixin");
traceHttp();
$wechatObj = new wechatCallbackapiTest();
if (isset($_GET['echostr'])) {
    $wechatObj->valid();
}else{
    $wechatObj->responseMsg();
}
```

当我们提交URL和Token验证的时候，程序目录下应当生成一个log.xml的文件。内容类似如下：

```
2014-01-10 11:03:21
REMOTE_ADDR:101.226.61.144
QUERY_STRING:signature=6e35c6f3d3279338781047dbffd09426b9ecdee3&echostr=5979420653038092664&timestamp=1392001400&nonce=1392192345
```

下面可以得出初步结论。

· 如果没有生成日志文件：微信服务器没有访问到服务器，你需要先检查一下你的服务器是否可以通过公网访问，以及URL路径是否正确并且可以访问，如果公网访问而微信服务器不能访问，那可能是防火墙拦截了80端口或微信服务器IP，也可能服务器所在区域与微信服务器通信不畅，需要更换服务器。

· 如果已经生成日志文件：查看REMOTE_ADDR和QUERY_STRING内容是否和上述类似。确认signature、timestamp、nonce、echostr四个参数是否都有值。如果这些都没有问题，则检查程序中定义的Token值是否和提交的一致，再检查程序流程及数据处理是否和官方文档描述的一致。

如果确定以上均没有问题，可以使用下面章节中的微信调试器进行测试，它提供了更为宽松的校验方式，并且可以实时输出当前的XML数据供调试时参考。

2.2.4 自动回复当前时间

在上面的例子中，我们已经嵌入了一个简单的时间查询功能，发送一个问号“？”就能回复当前的时间，如图2-26所示。

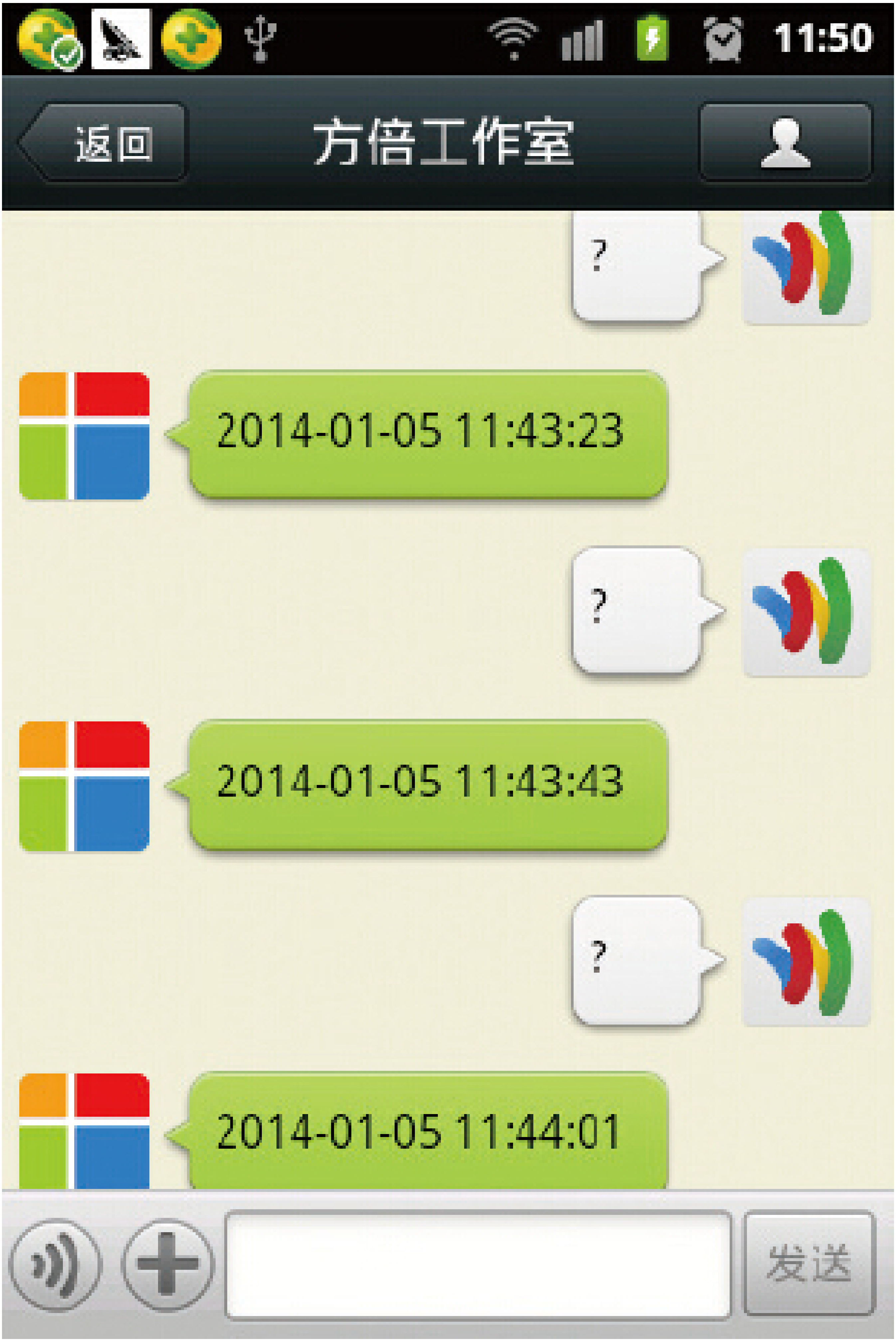

11:50
返回
方倍工作室
?
2014-01-05 11:43:23
?
2014-01-05 11:43:43
?
2014-01-05 11:44:01
发送

图2-26　自动回复时间

这个功能是基于下面的代码实现的。

```
if($keyword == "?" || $keyword == "？")
{
    $msgType = "text";
    $content = date("Y-m-d H:i:s",time());
    $result = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, $content);
    echo $result;
}
```

上述代码在收到消息后，判断消息内容是否为问号（包括英文输入状态下的问号和中文输入状态下的问号），如果是问号，则将当前时间（包括年、月、日、时、分、秒）作为回复内容，构造成一个消息回复过来。这样公众账号就实现了当前时间的自动回复。

2.2.5　消息交互原理分析

现在我们结合上一节的代码来分析一下微信公众平台的消息交互原理。下面的代码是基于微信公众平台官方示例代码修改完善而成的。

```
 1 <?php
 2 /*
 3     方倍工作室 http://www.fangbei.org/
 4     CopyRight 2013 www.doucube.com  All Rights Reserved
 5 */
 6
 7 define("TOKEN", "weixin");
 8 $wechatObj = new wechatCallbackapiTest();
 9 if (isset($_GET['echostr'])) {
10     $wechatObj->valid();
11 }else{
12     $wechatObj->responseMsg();
13 }
14
15 class wechatCallbackapiTest
16 {
17     public function valid()
18     {
19         $echoStr = $_GET["echostr"];
20         if($this->checkSignature()){
21             echo $echoStr;
22             exit;
23         }
24     }
25
26     private function checkSignature()
27     {
28         $signature = $_GET["signature"];
29         $timestamp = $_GET["timestamp"];
30         $nonce = $_GET["nonce"];
31
32         $token = TOKEN;
33         $tmpArr = array($token, $timestamp, $nonce);
34         sort($tmpArr);
35         $tmpStr = implode( $tmpArr );
36         $tmpStr = sha1( $tmpStr );
37
38         if( $tmpStr == $signature ){
39             return true;
40         }else{
41             return false;
42         }
43     }
44
45     public function responseMsg()
46     {
47         $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
48
49         if (!empty($postStr)){
50             $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
51             $fromUsername = $postObj->FromUserName;
52             $toUsername = $postObj->ToUserName;
53             $keyword = trim($postObj->Content);
54             $time = time();
55             $textTpl = "<xml>
56                         <ToUserName><![CDATA[%s]]></ToUserName>
57                         <FromUserName><![CDATA[%s]]></FromUserName>
58                         <CreateTime>%s</CreateTime>
59                         <MsgType><![CDATA[%s]]></MsgType>
60                         <Content><![CDATA[%s]]></Content>
61                         <FuncFlag>0</FuncFlag>
62                         </xml>";
63             if($keyword == "?" || $keyword == "？")
64             {
65                 $msgType = "text";
66                 $content = date("Y-m-d H:i:s",time());
67                 $result = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, $content);
68                 echo $result;
69             }
70         }else{
71             echo "";
72             exit;
73         }
74     }
75 }
76 ?>
```

首先我们来看一下代码的结构。

第2行~第5行是注释部分。

第7行使用define（）函数定义常量，常量名称为TOKEN，常量的值为weixin，这个值就是在启用开发模式时填写的Token。

第15行~第75行定义了一个类wechatCallbackapiTest，并在类中定义了3个方法valid（）、checkSignature（）、responseMsg（）。

第8行~第13行为程序执行语句。第8行实例化了一个类对象。在第9行中，判断GET请求是否设置了echostr变量，如果有则执行valid（）方法，否则执行responseMsg（）方法。

然后分析微信消息交互流程，具体如下。

在提交URL和Token申请验证的时候，微信服务器将发送GET请求到填写的URL上，并且带上四个参数（signature、timestamp、nonce、echostr），GET请求类似如下：

```
signature=6e35c6f3d3279338781047dbffd09426b9ecdee3&echostr=5979420653038092664&
timestamp=1392001400&nonce=1392192345
```

上述请求参数说明如表2-1所示。

表2-1　请求校验参数说明

参数	说明
signature	微信加密签名，signature 结合了开发者填写的 token 参数和请求中的 timestamp 参数、nonce 参数
timestamp	时间戳
nonce	随机数
echostr	随机字符串

这个GET请求是包含echostr变量的，所以执行valid（）方法，在该方法中，又调用了校验签名方法checkSignature（）方法。如果签名校验为真，则原样输出变量$echoStr的值。

加密/校验流程如下：

1）第33行~第34行将token、timestamp、nonce三个参数进行字典序排序。

2）第35行~第36行将三个参数字符串拼接成一个字符串进行sha1加密。

3）第38行~第42行表示开发者获得加密后的字符串可与signature对比，标识该请求来源于微信。

在发送问号的时候，微信服务器也将会带上前面三个参数（signature、timestamp、nonce）访问开发者设置的URL，同时还会将消息的XML数据包POST到URL上。XML格式类似如下：

```
<xml>
    <ToUserName><![CDATA[gh_ba6050bc0be7]]></ToUserName>
    <FromUserName><![CDATA[oDeOAjgSJUX10wvImSRMSwmyQAyA]]></FromUserName>
    <CreateTime>1392043637</CreateTime>
    <MsgType><![CDATA[text]]></MsgType>
    <Content><![CDATA[?]]></Content>
    <MsgId>5978781895719912033</MsgId>
</xml>
```

而消息请求不包含echostr变量，所以将执行响应消息responseMsg（）方法。

第47行表示响应消息方法首先接收上述原始POST数据。

第50行实现的功能是将数据载入对象中，对象名为SimpleXMLElement，LIBXML_NOCDATA表示将CDATA合并为文本节点。

第51行~第54行取得XML类对象的值，并赋予新的变量，注意发送方变为接收方，接收方变为发送方。

第55行~第62行构造要回复的XML数据包。

第63行判断发送过来的关键字是不是问号。

第64行~第65行设置回复的消息类型为text，内容为当前年、月、日、时、分、秒。

第66行~第67行封装回复的XML数据包，并且向微信服务器输出。XML格式如下所示：

```
<xml>
    <ToUserName><![CDATA[oDeOAjgSJUX10wvImSRMSwmyQAyA]]></ToUserName>
    <FromUserName><![CDATA[gh_ba6050bc0be7]]></FromUserName>
    <CreateTime>1392043638</CreateTime>
    <MsgType><![CDATA[text]]></MsgType>
    <Content><![CDATA[2014-01-05 11:43:23]]></Content>
</xml>
```

这样用户就会收到回复的消息，效果如图2-26所示。

2.3　微信开发调试工具

2.3.1　微信调试器

微信调试器是方倍工作室开发的用于微信公众平台接口开发在线调试的工具，具有Token校验、模拟关注及取消关注、发送文本/图片/语音/视频/位置/链接、模拟事件发送等功能。

微信调试器运行时推荐使用Chrome浏览器或Firefox浏览器，以得到更好的兼容。

微信调试器的地址是http://debug.fangbei.org/，其界面如图2-27所示。

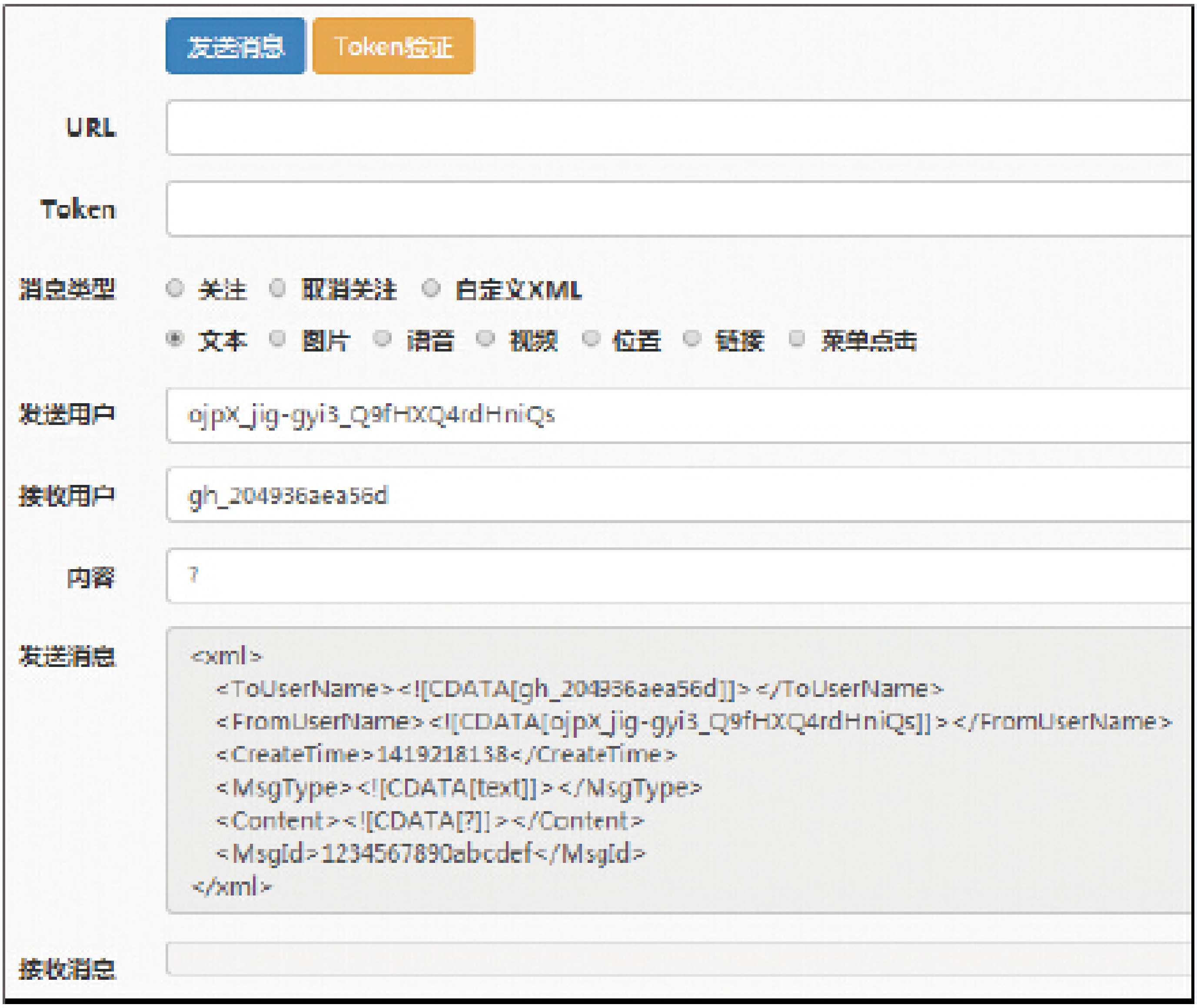


图2-27　微信调试器

下面介绍微信调试器的使用方法。

1.Token校验

在URL输入框和Token输入框中分别填写微信公众账号的接口URL和Token，这里的校验是明文模式的校验，不需要填写EncodingAESKey，如图2-28所示。

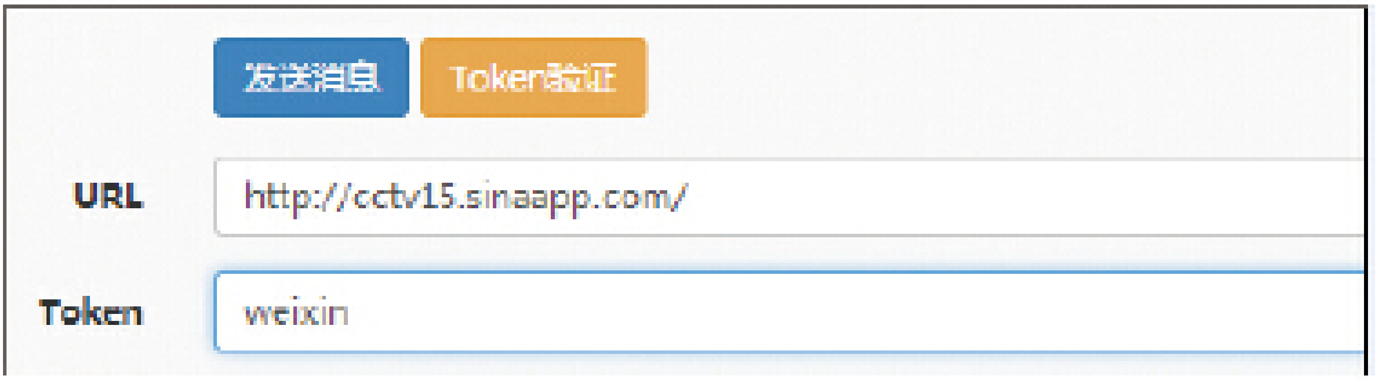


图2-28　Token校验

单击“Token验证”按钮，如果Token校验成功，将提示校验成功消息框，如图2-29所示；如果Token校验失败，将提示校验失败消息框。

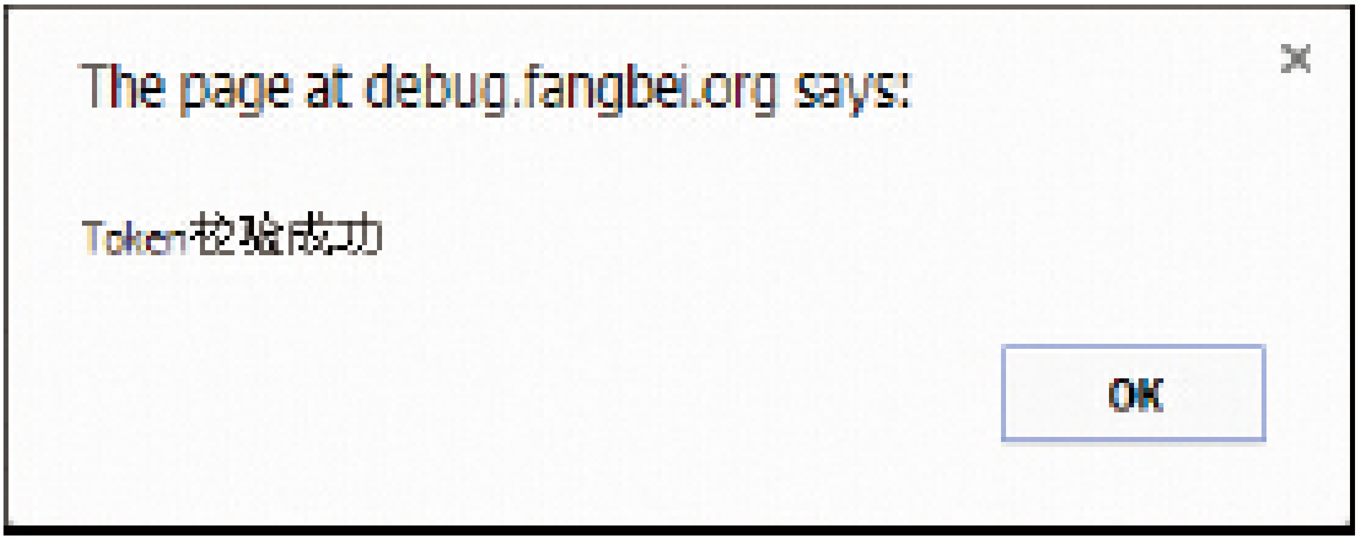


图2-29　Token校验成功

2.发送消息

选择消息类型，例如“文本”，将列出该消息类型的各项参数，在各项参数中填入要发送的参数内容（网站已经默认填充了一些固定的参数），然后单击“发送消息”按钮。发送消息框中将显示本次发送的XML，接收消息框中将会显示接收到的XML数据，如图2-30所示。

图2-30　发送消息

同时页面右侧也会显示微信效果预览图，如图2-31所示。

图2-31　微信效果预览图

如果接收消息框中没有返回XML或者返回的内容中包含非XML格式的数据，则说明返回不正确，需要修改自己的接口程序。

2.3　微信开发调试工具

2.3.1　微信调试器

微信调试器是方倍工作室开发的用于微信公众平台接口开发在线调试的工具，具有Token校验、模拟关注及取消关注、发送文本/图片/语音/视频/位置/链接、模拟事件发送等功能。

微信调试器运行时推荐使用Chrome浏览器或Firefox浏览器，以得到更好的兼容。

微信调试器的地址是http://debug.fangbei.org/，其界面如图2-27所示。

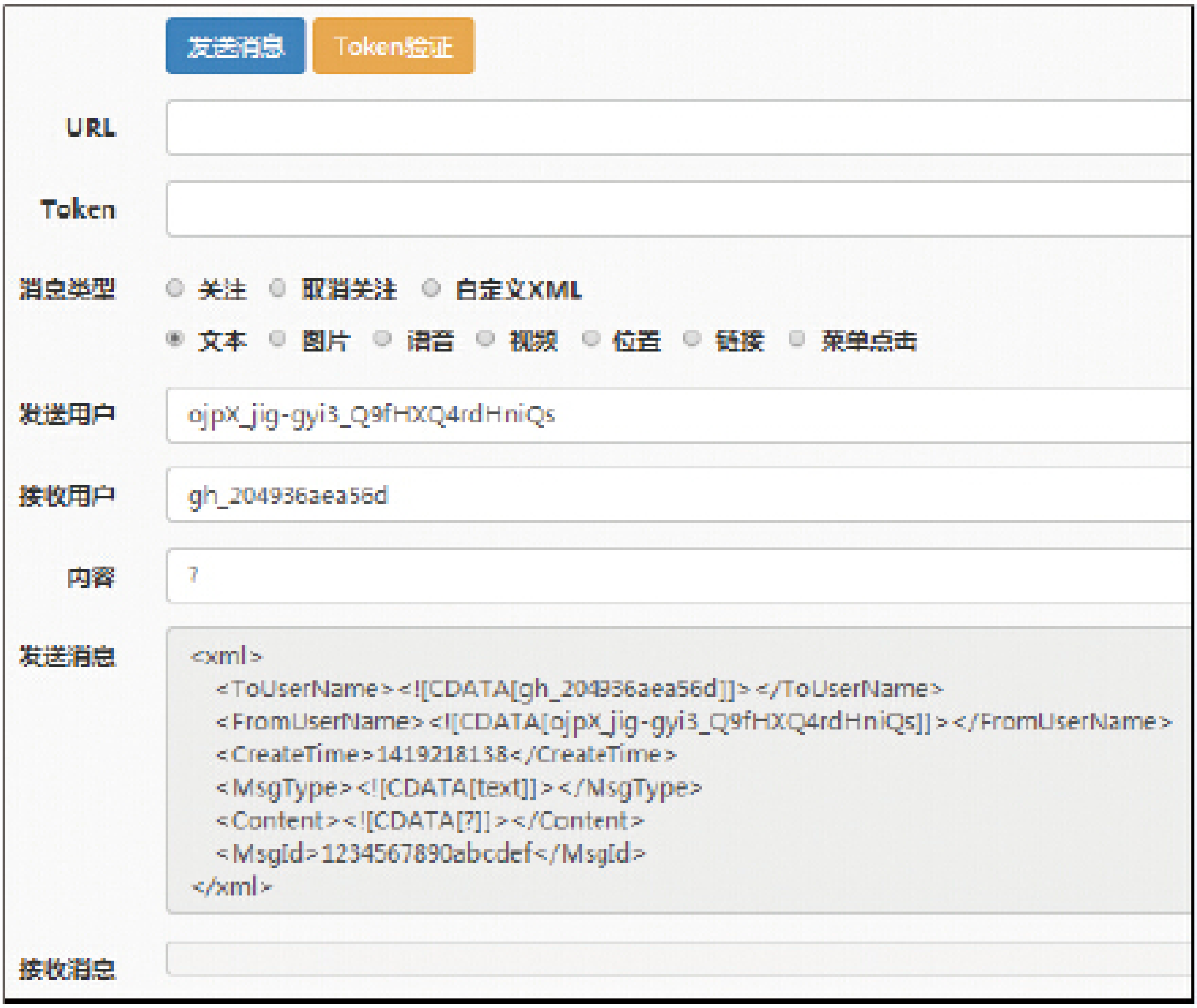


图2-27　微信调试器

下面介绍微信调试器的使用方法。

1.Token校验

在URL输入框和Token输入框中分别填写微信公众账号的接口URL和Token，这里的校验是明文模式的校验，不需要填写EncodingAESKey，如图2-28所示。

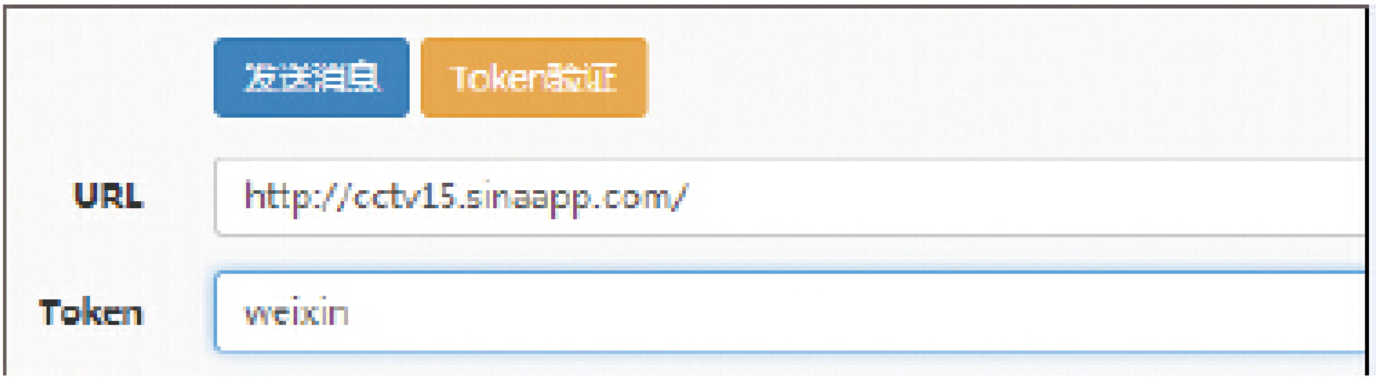


图2-28　Token校验

单击“Token验证”按钮，如果Token校验成功，将提示校验成功消息框，如图2-29所示；如果Token校验失败，将提示校验失败消息框。

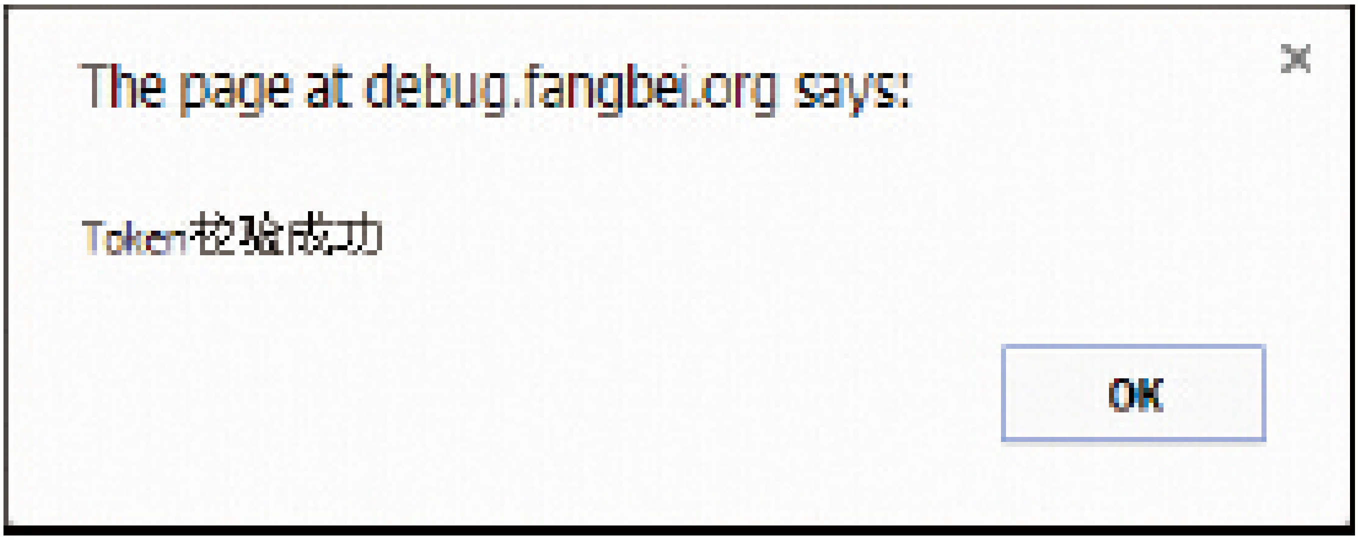


图2-29　Token校验成功

2.发送消息

选择消息类型，例如“文本”，将列出该消息类型的各项参数，在各项参数中填入要发送的参数内容（网站已经默认填充了一些固定的参数），然后单击“发送消息”按钮。发送消息框中将显示本次发送的XML，接收消息框中将会显示接收到的XML数据，如图2-30所示。

图2-30　发送消息

同时页面右侧也会显示微信效果预览图，如图2-31所示。

图2-31　微信效果预览图

如果接收消息框中没有返回XML或者返回的内容中包含非XML格式的数据，则说明返回不正确，需要修改自己的接口程序。

2.3.2　接口调试工具

微信公众平台提供了在线接口调试工具，网址为http://mp.weixin.qq.com/debug/。

该工具旨在帮助开发者检测调用微信公众平台开发者API时发送的请求参数是否正确，提交相关信息后可获得服务器的验证结果。在线接口调试工具如图2-32所示。

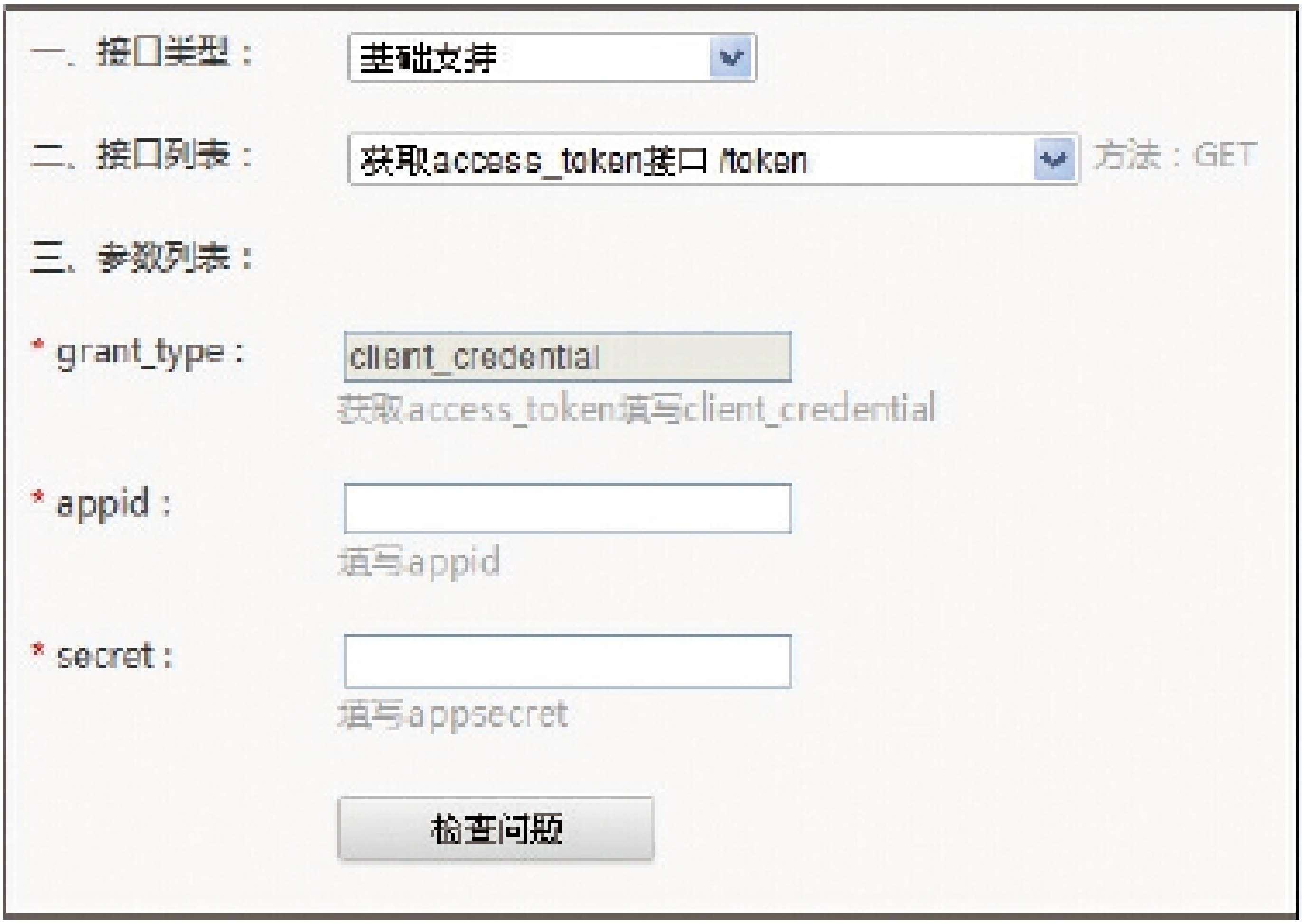


图2-32　在线接口调试工具

使用说明如下：

1）选择合适的接口。在接口类型和接口列表的下拉菜单中选择要调试的接口，如图2-33所示。

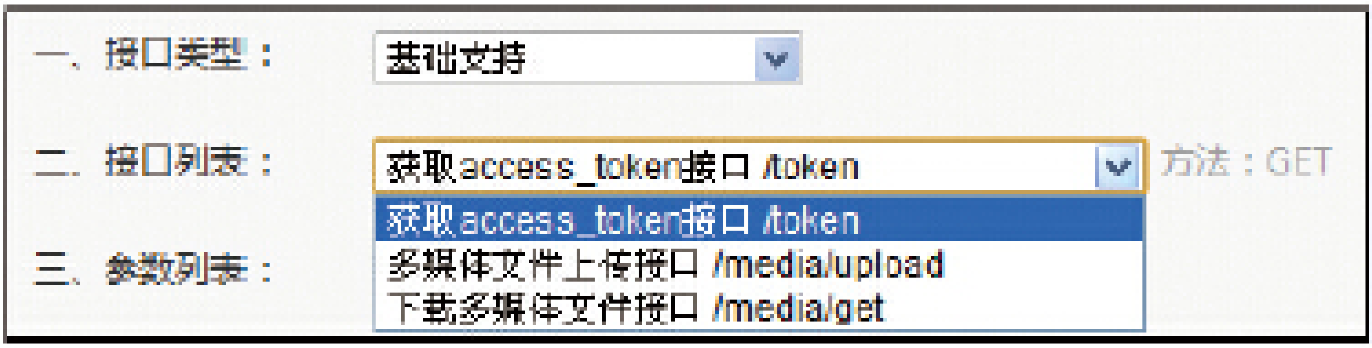


图2-33　选择接口

2）系统会生成该接口的参数表，您可以直接在文本框内填入对应的参数值，如图2-34所示（红色星号表示该字段必填）。

图2-34　填写参数

3）单击“检查问题”按钮，即可得到相应的调试信息，如图2-35所示。

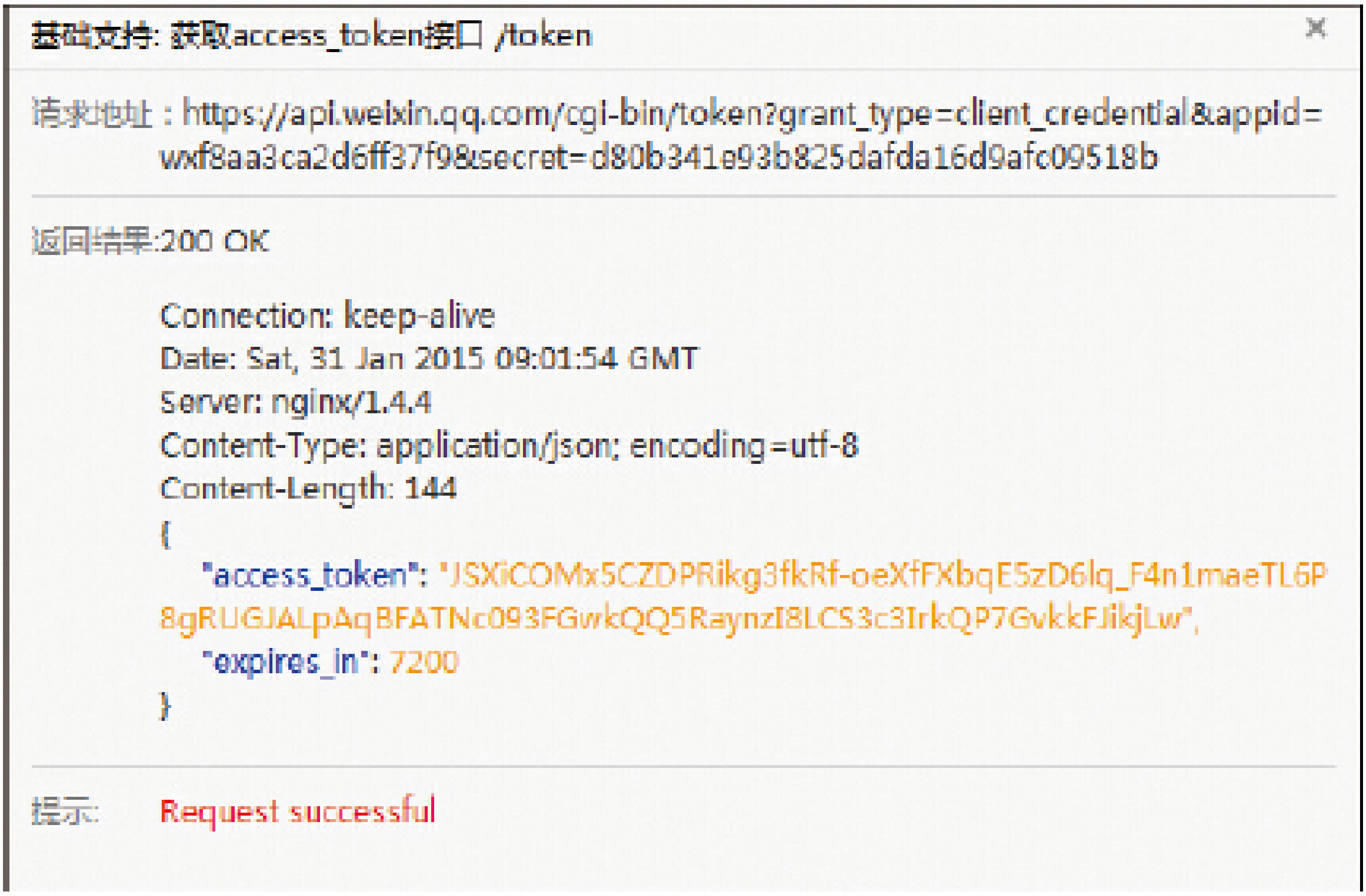


图2-35　调试结果信息

2.4　本章小结

本章首先介绍了如何申请新浪云SAE应用作为微信公众平台程序托管的服务器，然后介绍了配置及启用微信开发者中心的接口程序，最后介绍了使用微信调试器进行开发调试的方法。

需要说明的是，由于微信公众平台界面及SAE的界面经常进行改版，读者在实际操作时可能发现实际的界面和本书已经不完全一致，这种情况下，可以访问方倍工作室博客的《微信公众平台开发入门教程》获得最新的操作指导。地址是http://www.cnblogs.com/txw1958/p/wechat-tutorial.html。

第3章　基础接口

基础接口是微信公众账号在申请后就默认拥有而不需要通过微信认证获得的接口。微信公众平台开发者中心启用之后，需要使用基础接口实现基本消息的接收与发送。基础接口包括三个主要的部分：接收用户消息，发送被动响应消息，接收事件推送消息。本章将介绍这三个部分是如何实现的。

3.1　接收用户消息

微信公众账号能够接收用户发送的六种格式的消息：文本（包括表情）、图片、语音、视频、地理位置、链接。下面将分别介绍这六种消息。

3.1.1　文本消息

用户向微信公众账号发送文本消息的示例如图3-1所示。

17:22
返回
方倍工作室
下午17:21
微信公众平台开发教程
发送

图3-1　用户发送文本消息

用户发送文本消息时，微信公众账号接收到的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
    <FromUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_
ruZGsfk]]></FromUserName>
    <CreateTime>1359028446</CreateTime>
    <MsgType><![CDATA[text]]></MsgType>
    <Content><![CDATA[微信公众平台开发教程]]></Content>
    <MsgId>5836982729904121631</MsgId>
</xml>
```

用户发送的文本消息的参数及描述如表3-1所示。

表3-1　文本消息参数及描述

参数	描述
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	text
Content	文本消息内容
MsgId	消息 id，64 位整型

3.1.2　图片消息

用户向微信公众账号发送图片消息的示例如图3-2所示。

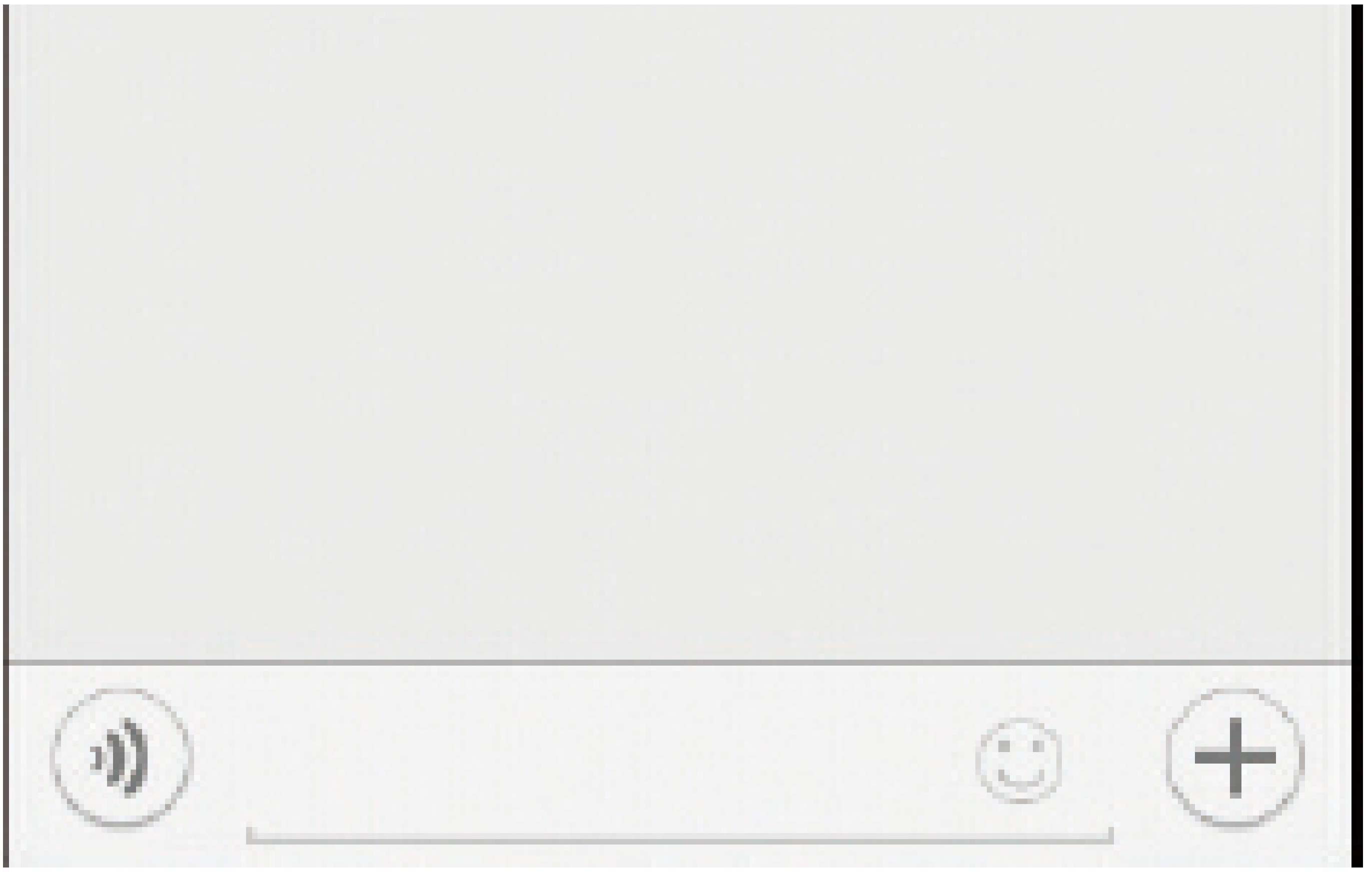


图3-2　用户发送图片消息

用户发送图片消息时，微信公众账号接收到的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
    <FromUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_ruZGsfk]]></FromUserName>
    <CreateTime>1359028479</CreateTime>
    <MsgType><![CDATA[image]]></MsgType>
    <PicUrl><![CDATA[http://mmbiz.qpic.cn/mmbiz/L4qjYtOibummHn90t1mnaibYiaR8ljyicF3MW7XX3BLp1qZgUb7CtZ0DxqYFI4uAQH1FWs3hUicpibjF0pOqLEQyDMlg/0]]></PicUrl>
    <MsgId>5836982871638042400</MsgId>
    <MediaId><![CDATA[PGKsO3LAgbVTsFYO7FGu51KUYa07D
0C_Nozz2fn1z6VYtHOsF59PTFl0vagGxkVH]]></MediaId>
</xml>
```

用户发送图片消息的参数及描述如表3-2所示。

表3-2　图片消息参数及描述

参数	描述
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	image
PicUrl	图片链接
MediaId	图片消息媒体 id，可以调用多媒体文件下载接口拉取数据
MsgId	消息 id，64 位整型

3.1.3　语音消息

用户向微信公众账号发送语音消息的示例如图3-3所示。

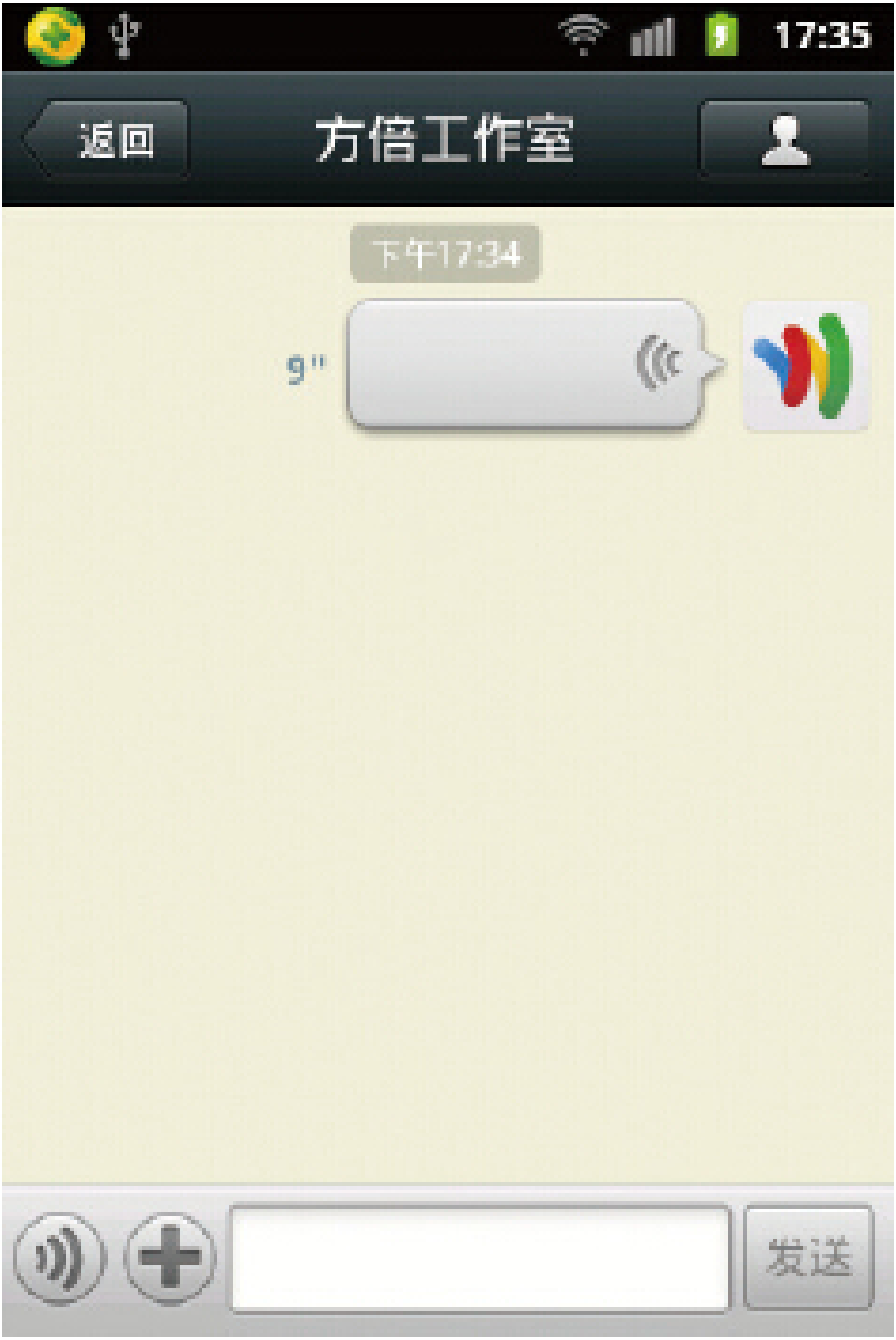

17:35
返回
方倍工作室
下午17:34
9"
发送

图3-3　用户发送语音消息

用户发送语音消息时，微信公众账号接收到的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
    <FromUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_
ruZGsfk]]></FromUserName>
    <CreateTime>1359028025</CreateTime>
    <MsgType><![CDATA[voice]]></MsgType>
    <MediaId><![CDATA[hGm9wmKth8RO_tuv5k9fJkSbovXWzZ
VYwG2jSsL7ukCqq6q1SiLzYnFEngFNUijs]]></MediaId>
    <Format><![CDATA[amr]]></Format>
    <MsgId>5836980921722890003</MsgId>
    <Recognition><![CDATA[]]></Recognition>
</xml>
```

用户发送语音消息的参数及描述如表3-3所示。

表3-3　语音消息参数及描述

参数	描述
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	voice
MediaId	语音消息媒体 id，可以调用多媒体文件下载接口拉取数据
Format	语音格式，如 amr、speex 等
MsgID	消息 id，64 位整型

3.1.4　视频消息

用户向微信公众账号发送视频消息的示例如图3-4所示。

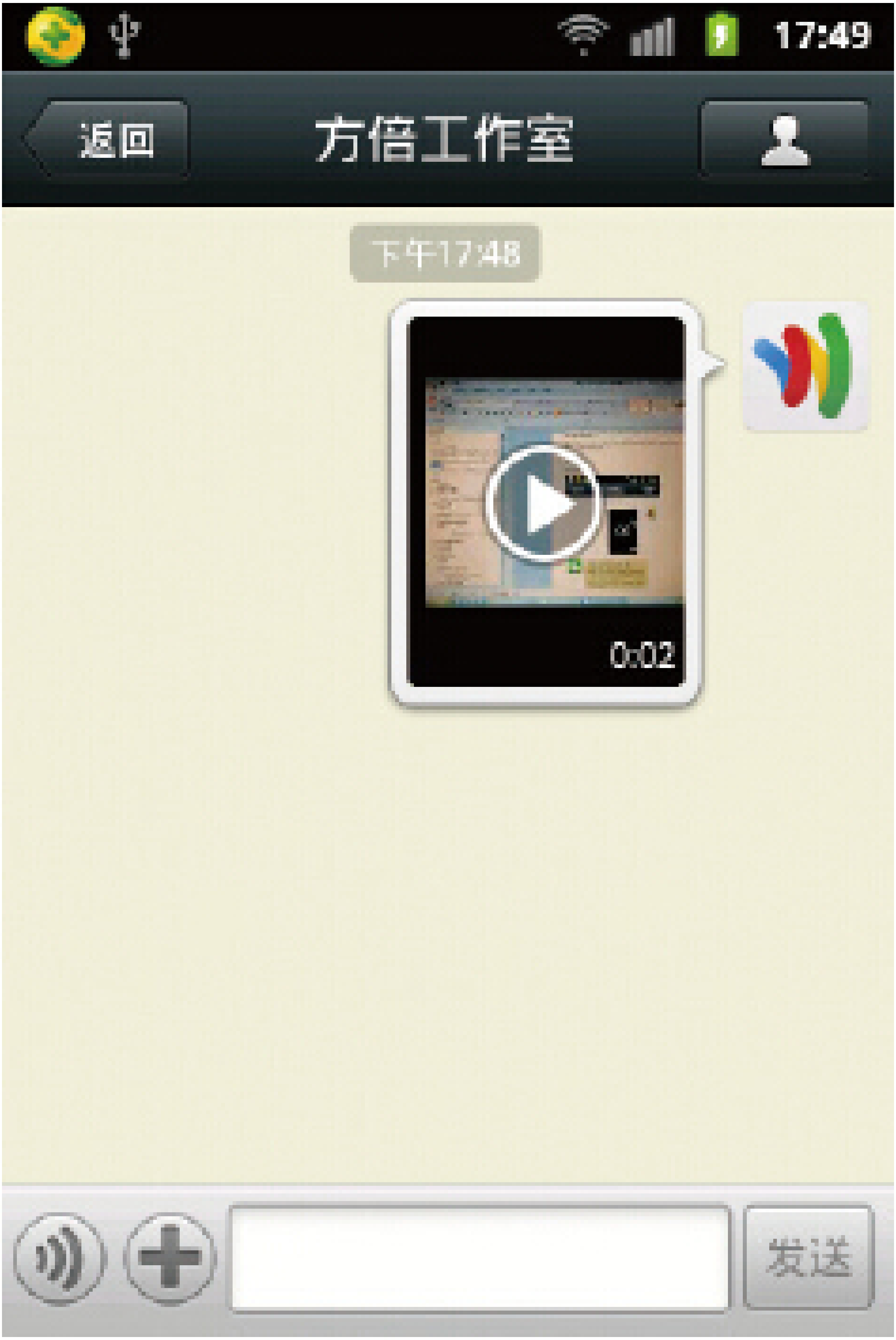

17:49
返回
方倍工作室
下午17:48
0:02
发送

图3-4　用户发送视频消息

用户发送视频消息时，微信公众账号接收到的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
    <FromUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_
ruZGsfk]]></FromUserName>
    <CreateTime>1359028186</CreateTime>
    <MsgType><![CDATA[video]]></MsgType>
    <MediaId><![CDATA[DBVFRIj29LB2hxuYpc0R6VLyxwgyCH
ZPbRj_IIs6YaGhutyXUKtFSDcSCPeoqUYr]]></MediaId>
    <ThumbMediaId><![CDATA[mxUJ5gcCeesJwx2T9qsk62YzI
clCP_HnRdfTQcojlPeT2G9Q3d22UkSLyBFLZ01J]]></ThumbMediaId>
    <MsgId>5836981613212624665</MsgId>
</xml>
```

用户发送视频消息的参数及描述如表3-4所示。

表3-4　视频消息参数及描述

参数	描述
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	video
MediaId	视频消息媒体 id，可以调用多媒体文件下载接口拉取数据
ThumbMediaId	视频消息缩略图的媒体 id，可以调用多媒体文件下载接口拉取数据
MsgId	消息 id，64 位整型

3.1.5　地理位置消息

用户向微信公众账号发送地理位置消息的示例如图3-5所示。

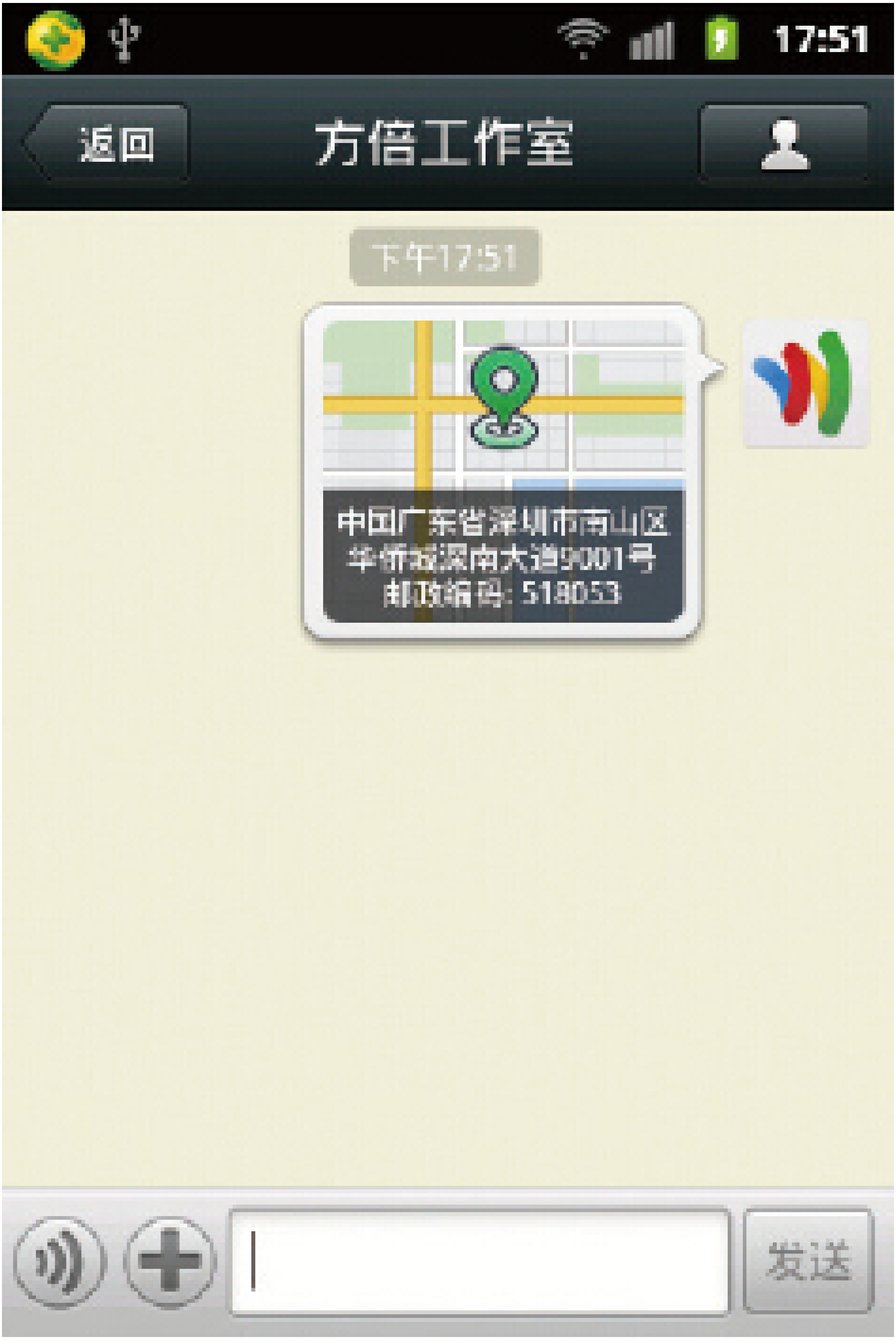

17:51
返回
方倍工作室
下午17:51
中国广东省深圳市南山区
华侨城深南大道9001号
邮政编码: 518053
发送

图3-5　用户发送地理位置消息

用户发送地理位置消息时，微信公众账号接收到的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
    <FromUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_
ruZGsfk]]></FLACFromUserName>
    <CreateTime>1359036619</CreateTime>
    <MsgType><![CDATA[location]]></MsgType>
    <Location_X>22.539968</Location_X>
    <Location_Y>113.954980</Location_Y>
    <Scale>16</Scale>
    <Label><![CDATA[中国广东省深圳市南山区深南大道9001号
邮政编码：518053]]></Label>
    <MsgId>5837017832671832047</MsgId>
</xml>
```

用户发送地理位置消息的参数及描述如表3-5所示。

表3-5　地理位置消息参数及描述

参数	描述	参数	描述
ToUserName	接收方微信号	Location_Y	地理位置经度
FromUserName	发送方账号（一个 OpenID）	Scale	地图缩放大小
CreateTime	消息创建时间（整型）	Label	地理位置信息
MsgType	location	MsgId	消息 id，64 位整型
Location_X	地理位置纬度		

3.1.6　链接消息

用户向微信公众账号发送链接消息的示例如图3-6所示。

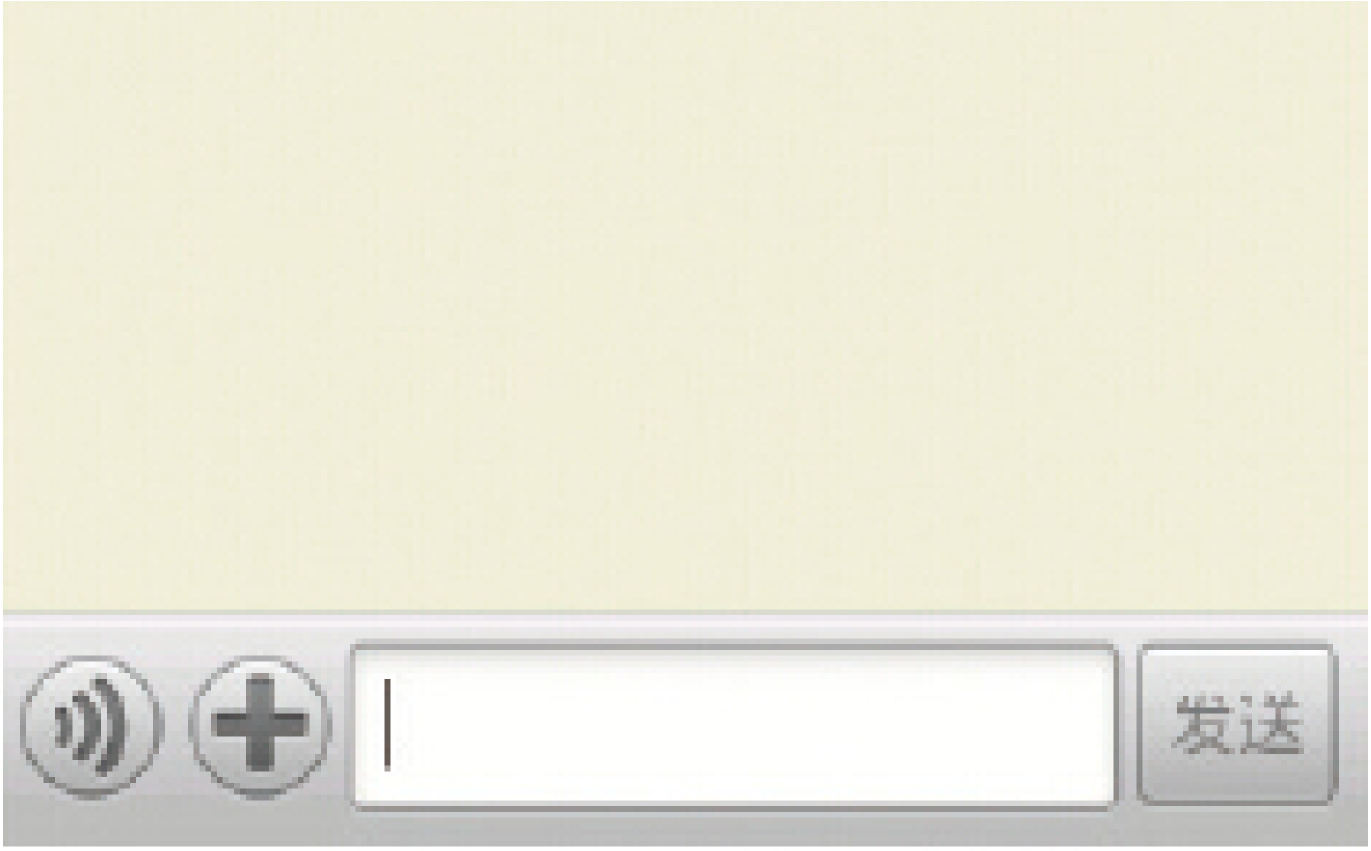


图3-6　用户发送链接消息

用户发送链接消息时，微信公众账号接收到的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_680bdefc8c5d]]></ToUserName>
    <FromUserName><![CDATA[oIDrpjl2LYdfTAM-oxDgB4XZ
cnc8]]></FromUserName>
    <CreateTime>1359709372</CreateTime>
    <MsgType><![CDATA[link]]></MsgType>
    <Title><![CDATA[微信公众平台开发者的江湖]]></Title>
    <Description><![CDATA[陈坤的微信公众号这段时间大火，大家http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/...]]></Descript
    <Url><![CDATA[http://www.cnblogs.com/txw1958/]]></Url>
    <MsgId>5839907284805129867</MsgId>
</xml>
```

用户发送的链接消息的参数及描述如表3-6所示。

表3-6　链接消息参数及描述

参数	描述	参数	描述
ToUserName	接收方微信号	Title	消息标题
FromUserName	发送方账号（一个 OpenID）	Description	消息描述
CreateTime	消息创建时间（整型）	Url	消息链接
MsgType	link	MsgId	消息 id，64 位整型

3.1.7　案例

前面的章节已经介绍了各种基础消息，本节将用代码实现上面的各种消息类型并且做出相应的回复处理。代码如下所示：

```
<?php
//
// 接收用户消息
// 微信公众账号接收到用户的消息类型判断
//
define("TOKEN", "weixin");
$wechatObj = new wechatCallbackapiTest();
if (!isset($_GET['echostr'])) {
    $wechatObj->responseMsg();
}else{
    $wechatObj->valid();
}
class wechatCallbackapiTest
{
    public function valid()
    {
        $echoStr = $_GET["echostr"];
        if($this->checkSignature()){
            echo $echoStr;
            exit;
        }
    }
    private function checkSignature()
```

```
    {
        $signature = $_GET["signature"];
        $timestamp = $_GET["timestamp"];
        $nonce = $_GET["nonce"];
        $token = TOKEN;
        $tmpArr = array($token, $timestamp, $nonce);
        sort($tmpArr);
        $tmpStr = implode( $tmpArr );
        $tmpStr = sha1( $tmpStr );
        if( $tmpStr == $signature ){
            return true;
        }else{
            return false;
        }
    }
    public function responseMsg()
    {
        $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
        if (!empty($postStr)){
            $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
            $RX_TYPE = trim($postObj->MsgType);
            //用户发送的消息类型判断
            switch ($RX_TYPE)
            {
                case "text":    //文本消息
                    $result = $this->receiveText($postObj);
                    break;
                case "image":   //图片消息
                    $result = $this->receiveImage($postObj);
                    break;
                case "voice":   //语音消息
                    $result = $this->receiveVoice($postObj);
                    break;
                case "video":   //视频消息
                    $result = $this->receiveVideo($postObj);
                    break;
                case "location"://位置消息
                    $result = $this->receiveLocation($postObj);
                    break;
                case "link":    //链接消息
                    $result = $this->receiveLink($postObj);
                    break;
                default:
                    $result = "unknow msg type: ".$RX_TYPE;
                    break;
            }
            echo $result;
        }else {
            echo "";
            exit;
        }
    }
    /*
     * 接收文本消息
     */
    private function receiveText($object)
    {
        $content = "你发送的是文本，内容为：".$object->Content;
        $result = $this->transmitText($object, $content);
        return $result;
    }
    /*
     * 接收图片消息
     */
    private function receiveImage($object)
    {
        $content = "你发送的是图片，地址为：".$object->PicUrl;
        $result = $this->transmitText($object, $content);
        return $result;
    }
    /*
     * 接收语音消息
     */
    private function receiveVoice($object)
    {
        $content = "你发送的是语音，媒体ID为：".$object->MediaId;
        $result = $this->transmitText($object, $content);
        return $result;
    }
    /*
     * 接收视频消息
     */
    private function receiveVideo($object)
    {
        $content = "你发送的是视频，媒体ID为：".$object->MediaId;
        $result = $this->transmitText($object, $content);
        return $result;
    }
    /*
     * 接收位置消息
     */
    private function receiveLocation($object)
    {
        $content = "你发送的是位置，纬度为：".$object->Location_X."；经度为：".$object->Location_Y."；缩放级别为：".$object->Scale."；位置为：".$object->Label;
        $result = $this->transmitText($object, $content);
        return $result;
    }
    /*
     * 接收链接消息
     */
    private function receiveLink($object)
    {
        $content = "你发送的是链接，标题为：".$object->Title."；内容为：".$object->
Description."；链接地址为：".$object->Url;
        $result = $this->transmitText($object, $content);
        return $result;
    }
    /*
     * 回复文本消息
     */
    private function transmitText($object, $content)
    {
        $textTpl = "<xml>
        <ToUserName><![CDATA[%s]]></ToUserName>
        <FromUserName><![CDATA[%s]]></FromUserName>
        <CreateTime>%s</CreateTime>
        <MsgType><![CDATA[text]]></MsgType>
        <Content><![CDATA[%s]]></Content>
        </xml>";
        $result = sprintf($textTpl, $object->FromUserName, $object-> ToUserName, time(), $content);
        return $result;
    }
}
?>
```

在responseMsg（）方法中，先提取消息类型$postObj->MsgType，从而实现各种消息类型的分离。在类wechatCallbackapiTest中，为每种消息类型定义了接收方法函数。在每个方法里面，返回消息的主要特征值，组成文本消息作为内容回复。

3.2 发送被动响应消息

微信公众账号能够回复用户发送的六种类型的消息：文本、图片、语音、视频、音乐、图文。其中图文又可分为单图文和多图文。下面将分别介绍这几种消息。

3.2.1　文本消息

微信公众账号向用户回复文本消息的示例如图3-7所示。

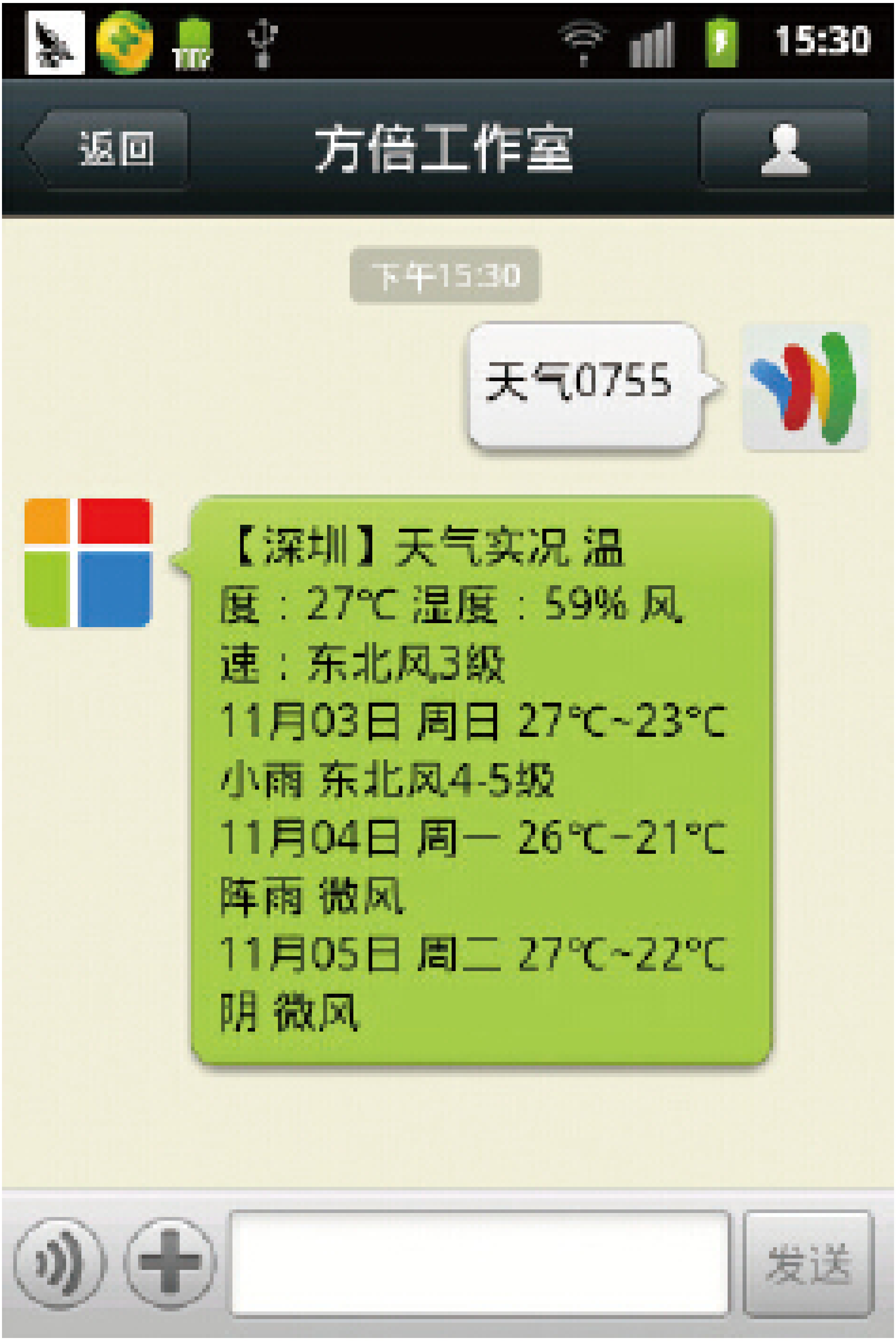

15:30
返回
方倍工作室
下午15:30
天气0755
【深圳】天气实况 温度：27℃ 湿度：59% 风速：东北风3级
11月03日 周日 27℃~23℃
小雨 东北风4-5级
11月04日 周一 26℃~21℃
阵雨 微风
11月05日 周二 27℃~22℃
阴 微风
发送

图3-7　公众账号回复文本消息

微信公众账号回复用户文本消息时的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_
ruZGsfk]]></ToUserName>
    <FromUserName><![CDATA[gh_680bdefc8c5d]]></From
UserName>
    <CreateTime>1359036631</CreateTime>
    <MsgType><![CDATA[text]]></MsgType>
    <Content><![CDATA[【深圳】天气实况 温度：27℃ 湿度：59% 风速：东北风3级
 11月03日 周日 27℃~23℃ 小雨 东北风4-5级 11月04日 周一 26℃~21℃ 阵雨 微风 11月05日 周二 27℃~22℃ 阴 微风]]></Content>
</xml>
```

公众账号回复文本消息的参数及描述如表3-7所示。

表3-7　文本消息参数及描述

参数	是否必须	描述
ToUserName	是	接收方账号（收到的 OpenID）
FromUserName	是	发送方微信号
CreateTime	是	消息创建时间（整型）
MsgType	是	text
Content	是	回复的消息内容

3.2.2　图片消息

微信公众账号向用户回复图片消息的示例如图3-8所示。

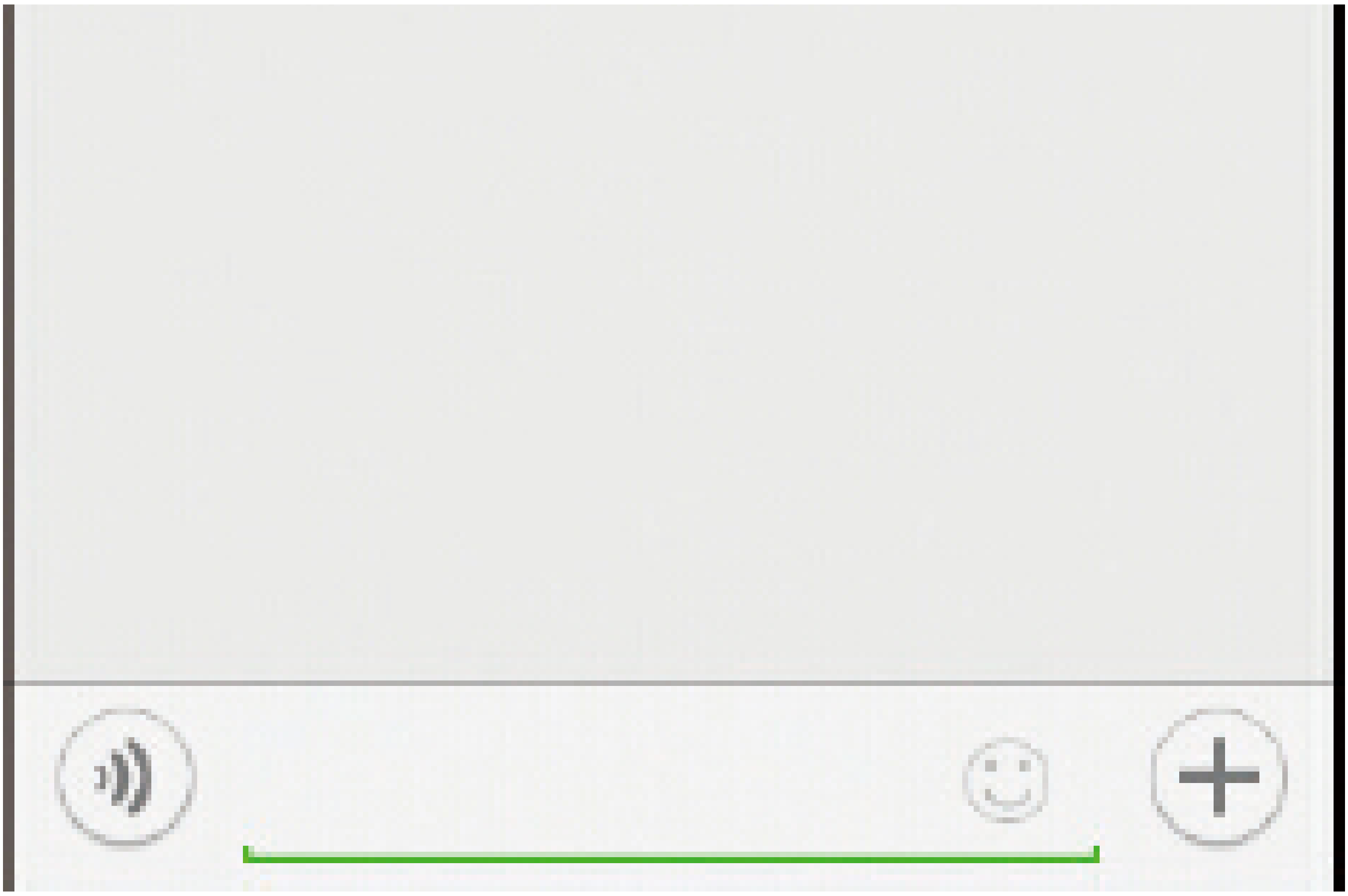


图3-8　公众账号回复图片消息

微信公众账号回复用户图片消息时的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[oDeOAjj54GvEkkgCV2d7QV4-
JLMc]]></ToUserName>
    <FromUserName><![CDATA[gh_ba6050bc0be7]]></From
UserName>
    <CreateTime>1392133855</CreateTime>
    <MsgType><![CDATA[image]]></MsgType>
    <Image>
    <MediaId><![CDATA[huNJ_LxG8vmFunz2Hjeb73X1IS02pu0jslBK24HAhSqi3bw2ZTCCYwKU2PaIer5n]]></MediaId>
    </Image>
</xml>
```

公众账号回复图片消息的参数及描述如表3-8所示。

表3-8　图片消息参数及描述

参数	是否必须	说明
ToUserName	是	接收方账号（收到的 OpenID）
FromUserName	是	发送方微信号
CreateTime	是	消息创建时间（整型）
MsgType	是	image
MediaId	是	通过上传多媒体文件得到的 id

3.2.3　语音消息

微信公众账号向用户回复语音消息的示例如图3-9所示。

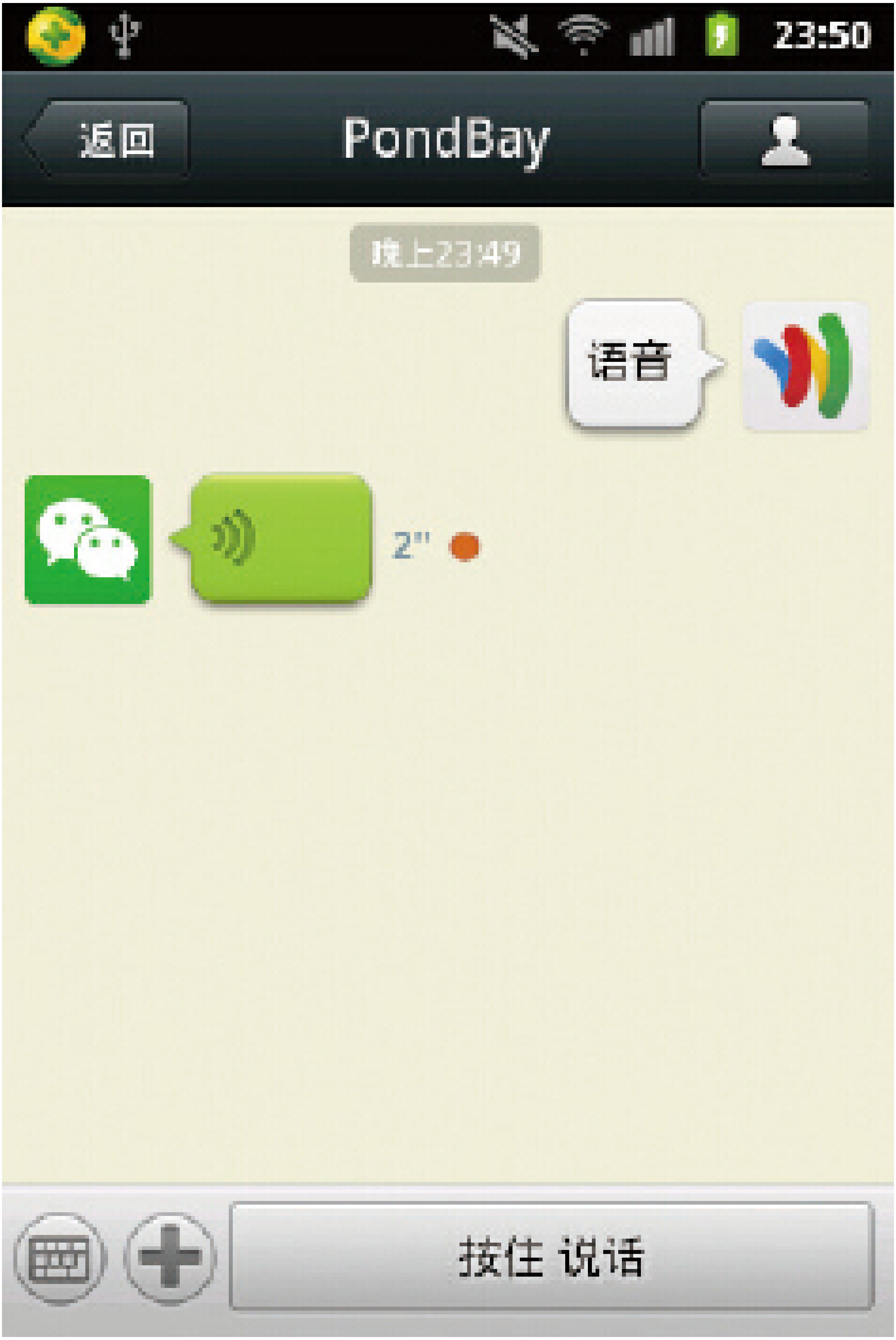

23:50
返回
PondBay
晚上23:49
语音
2''
按住 说话

图3-9　公众账号回复语音消息

微信公众账号回复用户语音消息时的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[oDeOAjj54GvEkkgCV2d7QV4-
JLMc]]></ToUserName>
    <FromUserName><![CDATA[gh_ba6050bc0be7]]></From
UserName>
    <CreateTime>1392133779</CreateTime>
    <MsgType><![CDATA[voice]]></MsgType>
    <Voice>
    <MediaId><![CDATA[ZKqseDPkTJ4dttQqNm_UPzoIHImELr
otOYjyALGJcdRZ2XcMQ6drvVabf5Dyr_Yx]]></MediaId>
    </Voice>
</xml>
```

公众账号回复语音消息的参数及描述如表3-9所示。

表3-9　语音消息参数及描述

参数	是否必须	说明
ToUserName	是	接收方账号（收到的 OpenID）
FromUserName	是	发送方微信号
CreateTime	是	消息创建时间戳（整型）
MsgType	是	voice
MediaId	是	通过上传多媒体文件得到的 id

3.2.4　视频消息

微信公众账号向用户回复视频消息的示例如图3-10所示。

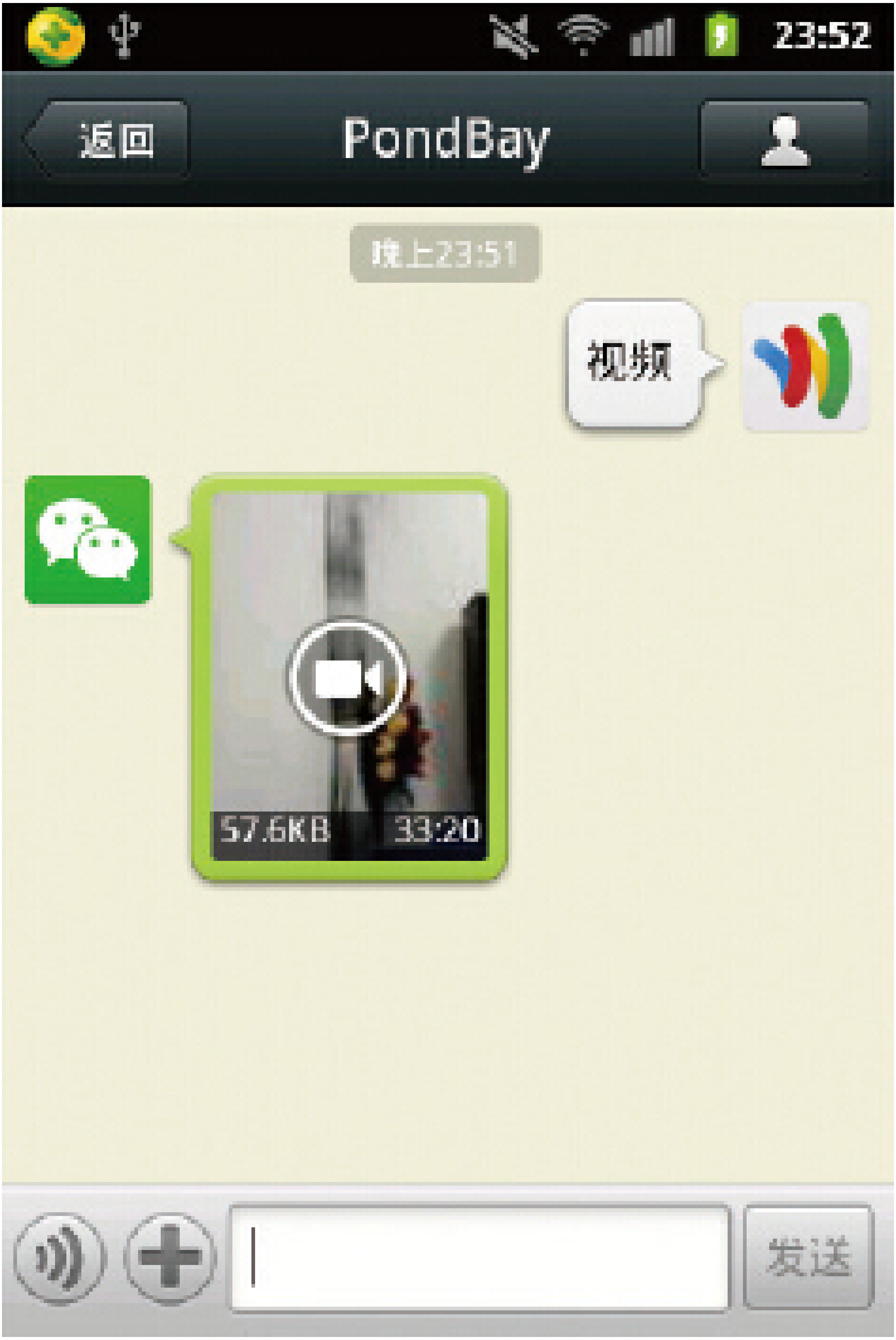

23:52
返回
PondBay
晚上23:51
视频
57.6KB
33:20
发送

图3-10　公众账号回复视频消息

微信公众账号回复用户视频消息的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[oDeOAjj54GvEkkgCV2d7QV4-
JLMc]]></ToUserName>
    <FromUserName><![CDATA[gh_ba6050bc0be7]]></From
UserName>
    <CreateTime>1392133911</CreateTime>
    <MsgType><![CDATA[video]]></MsgType>
    <Video>
    <MediaId><![CDATA[sFH7kkZ8I-9ioYPWwLzy47pg3AWXMR
4h0cr05asJdS8Pq3TlNTWpukrFjE-iPfgv]]></MediaId>
    <ThumbMediaId><![CDATA[9UMnGcFgaKD1ReW3c3gLerY-
c0zsrZtj0Vd6ZXWDmy9IYLyp-D5_blWTRU0pwihI]]></ThumbMediaId>
    <Title><![CDATA[Title]]></Title>
    <Description><![CDATA[Description]]></Description>
    </Video>
</xml>
```

公众账号回复视频消息的参数及描述如表3-10所示。

表3-10　视频消息参数及描述

参数	是否必须	说明
ToUserName	是	接收方账号（收到的 OpenID）
FromUserName	是	发送方微信号
CreateTime	是	消息创建时间（整型）
MsgType	是	video
MediaId	是	通过上传多媒体文件得到的 id
ThumbMediaId	是	缩略图的媒体 id，通过上传多媒体文件得到的 id
Title	否	视频消息的标题
Description	否	视频消息的描述

3.2.5　音乐消息

微信公众账号向用户回复音乐消息的示例如图3-11所示。

15:13
返回
方倍工作室
下午15:12
点歌最炫民族风
最炫民族风
凤凰传奇
发送

图3-11　公众账号回复音乐消息

微信公众账号回复用户音乐消息时的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[ollB4jqgdO_cRnVXk_wRnSywgtQ8]]></ToUserName>
    <FromUserName><![CDATA[gh_b629c48b653e]]></FromUserName>
    <CreateTime>1372310544</CreateTime>
    <MsgType><![CDATA[music]]></MsgType>
    <Music>
        <Title><![CDATA[最炫民族风]]></Title>
        <Description><![CDATA[凤凰传奇]]></Description>
        <MusicUrl><![CDATA[http://zj189.cn/zj/download/music/zxmzf.mp3]]></MusicUrl>
        <HQMusicUrl><![CDATA[http://zj189.cn/zj/download/music/zxmzf.mp3]]></HQMusicUrl>
    </Music>
</xml>
```

公众账号回复音乐消息的参数及描述如表3-11所示。

表3-11　音乐消息参数及描述

参数	是否必须	说明
ToUserName	是	接收方账号（收到的 OpenID）
FromUserName	是	发送方微信号
CreateTime	是	消息创建时间（整型）
MsgType	是	music
Title	否	音乐标题
Description	否	音乐描述
MusicURL	否	音乐链接
HQMusicUrl	否	高质量音乐链接，Wi-Fi 环境优先使用该链接播放音乐
ThumbMediaId	否	缩略图的媒体 id，通过上传多媒体文件得到的 id

3.2.6　图文消息

图文信息可以分为单图文和多图文，它们实现的代码是一样的，但在显示方式有细微区别。

微信公众账号向用户回复单图文消息的示例如图3-12所示。

图3-12　公众账号回复单图文消息

微信公众账号回复用户单图文消息的XML数据格式如下所示：

```xml
<xml>
    <ToUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_ruZGsfk]]></ToUserName>
    <FromUserName><![CDATA[gh_680bdefc8c5d]]></FromUserName>
    <CreateTime>1359011899</CreateTime>
    <MsgType><![CDATA[news]]></MsgType>
    <Content><![CDATA[]]></Content>
    <ArticleCount>1</ArticleCount>
    <Articles>
        <item>
            <Title><![CDATA[[苹果产品信息查询]]></Title>
            <Description><![CDATA[序列号：USE IMEI NUMBER
                                  ??IMEI号：358031058974471
                                  设备名称：iPhone 5C
                                  设备颜色：
                                  设备容量：
                                  激活状态：已激活
                                  电话支持：未过期[2014-01-13]
                                  硬件保修：未过期[2014-10-14]
                                  生产工厂：中国]]>
            </Description>
                <PicUrl><![CDATA[http://www.doucube.com/weixin/weather/icon/banner.jpg]]></PicUrl>
            <Url><![CDATA[]]></Url>
        </item>
    </Articles>
</xml>
```

微信公众账号向用户回复多图文消息的示例如图3-13所示。

图3-13　公众账号回复多图文消息

微信公众账号回复用户多图文消息时的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[oIDrpjqASyTPnxRmpS9O_ruZGsfk]]></ToUserName>
    <FromUserName><![CDATA[gh_680bdefc8c5d]]></FromUserName>
    <CreateTime>1359011829</CreateTime>
    <MsgType><![CDATA[news]]></MsgType>
    <Content><![CDATA[]]></Content>
    <ArticleCount>5</ArticleCount>
    <Articles>
        <item>
            <Title><![CDATA[  【深圳】实况 温度：6℃ 湿度：62% 风速：东北风2级]]> </Title>
            <Description><![CDATA[]]></Description>
        <PicUrl><![CDATA[http://www.doucube.com/weixin/weather/icon/banner.jpg]]></PicUrl>
            <Url><![CDATA[]]></Url>
        </item>
        <item>
        <Title><![CDATA[建议着厚外套加毛衣等服装。年老体弱者宜着大衣、呢外套加羊毛衫。]]></Title>
            <Description><![CDATA[]]></Description>
        <PicUrl><![CDATA[http://www.doucube.com/weixin/weather/icon/d00.gif]]></PicUrl>
            <Url><![CDATA[]]></Url>
        </item>
        <item>
            <Title><![CDATA[02月12日 周三 10℃~6℃ 阴 微风]]></Title>
            <Description><![CDATA[]]></Description>
        <PicUrl><![CDATA[http://www.doucube.com/weixin/weather/icon/d00.gif]]></PicUrl>
            <Url><![CDATA[]]></Url>
        </item>
        <item>
            <Title><![CDATA[02月13日 周四 11℃~7℃ 小雨 微风]]></Title>
            <Description><![CDATA[]]></Description>
        <PicUrl><![CDATA[http://www.doucube.com/weixin/weather/icon/d01.gif]]></PicUrl>
            <Url><![CDATA[]]></Url>
        </item>
        <item>
        <Title><![CDATA[02月14日 周五 11℃~8℃ 小雨 东北风3-4级]]></Title>
            <Description><![CDATA[]]></Description>
        <PicUrl><![CDATA[http://www.doucube.com/weixin/weather/icon/d01.gif]]></PicUrl>
            <Url><![CDATA[]]></Url>
        </item>
    </Articles>
</xml>
```

公众账号回复图文消息的参数及描述如表3-12所示。

表3-12　图文消息参数及描述

参数	是否必须	说明
ToUserName	是	接收方账号（收到的 OpenID）
FromUserName	是	发送方微信号
CreateTime	是	消息创建时间（整型）
MsgType	是	news
ArticleCount	是	图文消息个数，限制为 10 条以内
Articles	是	多条图文消息信息，默认第一个 item 为大图；注意，如果图文数超过 10，则将会无响应
Title	否	图文消息标题
Description	否	图文消息描述
PicUrl	否	图片链接，支持 JPG、PNG 格式，较好的效果为大图 360×200 像素，小图 200×200 像素
Url	否	单击图文消息跳转链接

在单图文消息中，标题、描述、图片分开显示在各处，图片为大图。在多图文信息中，每条信息将只显示标题内容，描述字段中的内容将不显示，第一条消息标题与图片层叠，显示在上方，从第二条消息开始，对应的图片显示为小图。

3.2.7 案例

本节将以上六种类型的响应消息通过代码实现，以便读者理解。代码如下所示：

```
<?php
//
// 响应用户消息
// 微信公众账号响应给用户的不同消息类型
//
define("TOKEN", "weixin");
$wechatObj = new wechatCallbackapiTest();
if (!isset($_GET['echostr'])) {
    $wechatObj->responseMsg();
}else{
    $wechatObj->valid();
}
class wechatCallbackapiTest
{
    public function valid()
    {
        $echoStr = $_GET["echostr"];
        if($this->checkSignature()){
            echo $echoStr;
            exit;
        }
    }
    private function checkSignature()
    {
        $signature = $_GET["signature"];
        $timestamp = $_GET["timestamp"];
        $nonce = $_GET["nonce"];
        $token = TOKEN;
        $tmpArr = array($token, $timestamp, $nonce);
        sort($tmpArr);
        $tmpStr = implode($tmpArr);
        $tmpStr = sha1($tmpStr);
        if($tmpStr == $signature){
            return true;
        }else{
            return false;
        }
    }
    public function responseMsg()
    {
        $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
        if (!empty($postStr)){
        $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
            $RX_TYPE = trim($postObj->MsgType);
            //用户发送的消息类型判断
            switch ($RX_TYPE)
            {
                case "text":
                    $result = $this->receiveText($postObj);
                    break;
                case "image":
                    $result = $this->receiveImage($postObj);
                    break;
                case "voice":
                    $result = $this->receiveVoice($postObj);
                    break;
                case "video":
                    $result = $this->receiveVideo($postObj);
                    break;
                default:
                    $result = "unknow msg type: ".$RX_TYPE;
                    break;
            }
            echo $result;
        }else {
            echo "";
            exit;
        }
    }
    private function receiveText($object)
    {
        $keyword = trim($object->Content);
        if($keyword == "文本"){
            //回复文本消息
            $content = "这是个文本消息";
            $result = $this->transmitText($object, $content);
        }
        else if($keyword == "图文" || $keyword == "单图文"){
            //回复单图文消息
            $content = array();
            $content[] = array("Title"=>"单图文标题",
                              "Description"=>"单图文内容",
        "PicUrl"=>"http://discuz.comli.com/weixin/weather/icon/cartoon.jpg",
        "Url" =>"http://m.cnblogs.com/?u= txw1958");
```

```
            $result = $this->transmitNews($object, $content);
        }
        else if($keyword == "多图文"){
            //回复多图文消息
            $content = array();
            $content[] = array("Title"=>"多图文1标题", "Description"=>"",
"PicUrl"=>"http://discuz.comli.com/weixin/weather/icon/cartoon.jpg", "Url" =>"http://m.cnblogs.com/?u=txw1958");
         $content[] = array("Title"=>"多图文2标题", "Description"=>"", "PicUrl"=>"http://d.hiphotos.bdimg.com/wisegame/pic/item/f3529822720e0cf3ac9f1ada0846f21fbe09aaa3.jpg", "U
         $content[] = array("Title"=>"多图文3标题", "Description"=>"", "PicUrl"=>"http://g.hiphotos.bdimg.com/wisegame/pic/item/18cb0a46f21fbe090d338acc6a600c338644adfd.jpg", "U
            $result = $this->transmitNews($object, $content);
        }
        else if($keyword == "音乐"){
            //回复音乐消息
            $content = array("Title"=>"最炫民族风",
            "Description"=>"歌手：凤凰传奇",
            "MusicUrl"=>"http://121.199.4.61/music/zxmzf.mp3",
            "HQMusicUrl"=>"http://121.199.4.61/music/zxmzf.mp3");
            $result = $this->transmitMusic($object, $content);
        }
        return $result;
    }
    private function receiveImage($object)
    {
        //回复图片消息
        $content = array("MediaId"=>$object->MediaId);
        $result = $this->transmitImage($object, $content);;
        return $result;
    }
    private function receiveVoice($object)
    {
        //回复语音消息
        $content = array("MediaId"=>$object->MediaId);
        $result = $this->transmitVoice($object, $content);;
        return $result;
    }
    private function receiveVideo($object)
    {
        //回复视频消息
        $content = array("MediaId"=>$object->MediaId, "ThumbMediaId"=> $object->ThumbMediaId, "Title"=>"", "Description"=>"");
        $result = $this->transmitVideo($object, $content);;
        return $result;
    }
    /*
     * 回复文本消息
     */
    private function transmitText($object, $content)
    {
        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[text]]></MsgType>
<Content><![CDATA[%s]]></Content>
</xml>";
        $result = sprintf($textTpl, $object->FromUserName, $object-> ToUserName, time(), $content);
        return $result;
    }
    /*
     * 回复图片消息
     */
    private function transmitImage($object, $imageArray)
    {
        $itemTpl = "<Image>
    <MediaId><![CDATA[%s]]></MediaId>
</Image>";
        $item_str = sprintf($itemTpl, $imageArray['MediaId']);
        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[image]]></MsgType>
$item_str
</xml>";
        $result = sprintf($textTpl, $object->FromUserName, $object-> ToUserName, time());
        return $result;
    }
    /*
     * 回复语音消息
     */
    private function transmitVoice($object, $voiceArray)
    {
        $itemTpl = "<Voice>
    <MediaId><![CDATA[%s]]></MediaId>
</Voice>";
        $item_str = sprintf($itemTpl, $voiceArray['MediaId']);
        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[voice]]></MsgType>
$item_str
</xml>";
        $result = sprintf($textTpl, $object->FromUserName, $object-> ToUserName, time());
        return $result;
    }
    /*
     * 回复视频消息
     */
    private function transmitVideo($object, $videoArray)
    {
        $itemTpl = "<Video>
    <MediaId><![CDATA[%s]]></MediaId>
    <ThumbMediaId><![CDATA[%s]]></ThumbMediaId>
    <Title><![CDATA[%s]]></Title>
    <Description><![CDATA[%s]]></Description>
</Video>";
        $item_str = sprintf($itemTpl, $videoArray['MediaId'], $videoArray
['ThumbMediaId'], $videoArray['Title'], $videoArray['Description']);
        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[video]]></MsgType>
$item_str
</xml>";
        $result = sprintf($textTpl, $object->FromUserName, $object-> ToUserName, time());
        return $result;
    }
    /*
     * 回复图文消息
     */
    private function transmitNews($object, $arr_item)
    {
        if(!is_array($arr_item))
            return;
        $itemTpl = "    <item>
        <Title><![CDATA[%s]]></Title>
        <Description><![CDATA[%s]]></Description>
        <PicUrl><![CDATA[%s]]></PicUrl>
        <Url><![CDATA[%s]]></Url>
    </item>
";
        $item_str = "";
        foreach ($arr_item as $item)
            $item_str .= sprintf($itemTpl, $item['Title'], $item ['Description'], $item['PicUrl'], $item['Url']);
        $newsTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
```

```
<MsgType><![CDATA[news]]></MsgType>
<Content><![CDATA[]]></Content>
<ArticleCount>%s</ArticleCount>
<Articles>
$item_str</Articles>
</xml>";
        $result = sprintf($newsTpl, $object->FromUserName, $object-> ToUserName, time(), count($arr_item));
        return $result;
    }
    /*
     * 回复音乐消息
     */
    private function transmitMusic($object, $musicArray)
    {
        $itemTpl = "<Music>
    <Title><![CDATA[%s]]></Title>
    <Description><![CDATA[%s]]></Description>
    <MusicUrl><![CDATA[%s]]></MusicUrl>
    <HQMusicUrl><![CDATA[%s]]></HQMusicUrl>
</Music>";
        $item_str = sprintf($itemTpl, $musicArray['Title'], $musicArray ['Description'], $musicArray['MusicUrl'], $musicArray['HQMusicUrl']);
        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[music]]></MsgType>
$item_str
</xml>";
        $result = sprintf($textTpl, $object->FromUserName, $object-> ToUserName, time());
        return $result;
    }
}
?>
```

在上面的代码中，回复文本、图文（包括单图文和多图文）、音乐三种消息，是使用直接构造响应消息类型实现的，而图片、语音、视频三种消息需要MediaId参数，在这里直接使用用户发送过来的消息中的MediaId，然后组装成响应消息回复。

3.3 接收事件推送消息

在基础接口中，事件消息只有关注和取消关注事件。用户关注和取消关注公众账号的时候将分别触发这两个消息。

3.3.1 关注/取消关注

用户关注微信公众账号时的界面如图3-14所示，单击“关注”按钮，微信公众账号将收到关注事件。

10:00
返回
详细资料
方倍工作室
功能介绍
微信公众平台综合解决方案提供商
查看历史消息
查看地理位置
关注

图3-14　关注微信公众账号

用户关注微信公众账号时的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_b629c48b653e]]></ToUserName>
    <FromUserName><![CDATA[ollB4jv7LA3tydjviJp5V9qTU_kA]]></FromUserName>
    <CreateTime>1372307736</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[subscribe]]></Event>
    <EventKey><![CDATA[]]></EventKey>
</xml>
```

用户取消关注微信公众账号时的界面如图3-15所示，单击“取消关注”按钮，微信公众账号将收到取消关注事件。

图3-15　取消关注微信公众账号

用户取消关注微信公众账号时的XML数据格式如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_b629c48b653e]]></ToUserName>
    <FromUserName><![CDATA[ollB4jqgdO_cRnVXk_wRnSywgtQ8]]></FromUserName>
    <CreateTime>1372309890</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[unsubscribe]]></Event>
    <EventKey><![CDATA[]]></EventKey>
</xml>
```

关注及取消关注事件消息的参数及描述如表3-13所示。

表3-13　文本消息参数及描述

参数	描述
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	event
Event	事件类型，subscribe(关注)、unsubscribe(取消关注)
Event	事件 KEY 值，关注 / 取消关注事件中为空

3.3.2　案例

本节将关注/取消关注事件消息通过代码实现，以便读者理解。代码如下所示：

```
<?php
//
// 关注/取消关注事件消息
// 微信公众账号关注与取消关注事件消息
//
define("TOKEN", "weixin");
$wechatObj = new wechatCallbackapiTest();
if (!isset($_GET['echostr'])) {
    $wechatObj->responseMsg();
}else{
    $wechatObj->valid();
}
class wechatCallbackapiTest
{
    public function valid()
    {
        $echoStr = $_GET["echostr"];
        if($this->checkSignature()){
            echo $echoStr;
            exit;
        }
    }
    private function checkSignature()
    {
        $signature = $_GET["signature"];
        $timestamp = $_GET["timestamp"];
        $nonce = $_GET["nonce"];
        $token = TOKEN;
        $tmpArr = array($token, $timestamp, $nonce);
        sort($tmpArr);
        $tmpStr = implode($tmpArr);
        $tmpStr = sha1($tmpStr);
        if($tmpStr == $signature){
            return true;
        }else{
            return false;
        }
    }
    public function responseMsg()
    {
        $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
        if (!empty($postStr)){
        $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
            $RX_TYPE = trim($postObj->MsgType);
            switch ($RX_TYPE)
            {
                case "event":
                    $result = $this->receiveEvent($postObj);
                    break;
            }
            echo $result;
        }else {
            echo "";
            exit;
        }
    }
    private function receiveEvent($object)
    {
        $content = "";
        switch ($object->Event)
        {
            case "subscribe":   //关注事件
                $content = "欢迎关注方倍工作室";
                break;
            case "unsubscribe": //取消关注事件
                $content = "";
                break;
        }
        $result = $this->transmitText($object, $content);
        return $result;
    }
    private function transmitText($object, $content)
    {
        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[text]]></MsgType>
<Content><![CDATA[%s]]></Content>
</xml>";
        $result = sprintf($textTpl, $object->FromUserName, $object-> ToUserName, time(), $content);
        return $result;
    }
```

```
}
?>
```

在上面的代码中，判断消息类型为event而进入事件方法，再判断事件类型是否为关注或取消关注事件：当为关注事件的时候，回复一条欢迎语；当事件类型为取消关注时，直接回复空消息。因为这时用户已经取消关注，接收不到任何消息了。

3.4　本章小结

本章介绍了微信公众平台的基础接口，包含接收用户消息、发送被动响应消息、接收事件推送消息等三部分，这些接口是所有公众账号都具备的基础能力，熟练地应用这些接口可以实现很多有趣、有价值的功能。比如图片消息中的人脸识别、地理位置消息中的导航功能，这些都将在后面的章节有介绍。

第4章　自定义菜单和高级接口

自定义菜单和高级接口都需要使用AppID和AppSecret来创建，为了方便读者更好地理解它们的原理及机制，将它们放在同一章讲解。对于暂时没有这些权限的微信公众账号，开发者可以申请测试账号来体验和测试体验微信公众平台的所有高级接口的功能。在第一章有测试账号的申请方法的介绍。

4.1　Access Token

4.1.1　Access Token概述

在微信公众平台接口开发中，Access Token占据了一个很重要的地位，相当于进入各种接口的钥匙，拿到这个钥匙才有调用其他各种特殊接口的权限。

access_token是微信公众账号的全局唯一凭证，微信公众账号调用各接口时都需使用access_token。正常情况下access_token有效期为7200秒，重复获取将导致上次获取的access_token失效。

微信公众账号可以使用AppID和AppSecret调用本接口来获取access_token。AppID和AppSecret可在开发模式中获得（需要已经成为开发者，且账号没有异常状态）。注意调用所有微信接口时均需使用HTTPS协议。

4.1　Access Token

4.1.1　Access Token概述

在微信公众平台接口开发中，Access Token占据了一个很重要的地位，相当于进入各种接口的钥匙，拿到这个钥匙才有调用其他各种特殊接口的权限。

access_token是微信公众账号的全局唯一凭证，微信公众账号调用各接口时都需使用access_token。正常情况下access_token有效期为7200秒，重复获取将导致上次获取的access_token失效。

微信公众账号可以使用AppID和AppSecret调用本接口来获取access_token。AppID和AppSecret可在开发模式中获得（需要已经成为开发者，且账号没有异常状态）。注意调用所有微信接口时均需使用HTTPS协议。

4.1.2　接口调用请求说明

获取Access Token的接口如下：

```
https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=APPID&secret=APPSECRET
```

Access Token接口的参数说明如表4-1所示。

表4-1　Access Token参数说明

参数	是否必须	说明
grant_type	是	获取 access_token 填写 client_credential
appid	是	第三方用户唯一凭证
secret	是	第三方用户唯一凭证密钥，即 appsecret

该接口的实例请求如下：

```
https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=wx9a000b615d89c3f1&secret=0ca77dd808ee1ea69830d7eecd770c06
```

执行上述请求后，接口返回如下：

```
{
    "access_token":
"Ul24w0iK4_-g5te8UqjU9CKw0Cf-Iks7wr8YGjsV3cO2wIJhZlwMLs8rtBiB-cJ3MdTeSgkYMbjtrfaDmUTsn2rEjldKd59HOdftaPxDTPmCj3uJBXGjNz9P6LkMsl384G6HSEh-uLgixB8gmgOveA",
    "expires_in": 7200
}
```

该返回结果参数说明如表4-2所示。

表4-2　Access Token返回结果参数说明

参数	说明
access_token	获取到的凭证
expires_in	凭证有效时间，单位：秒

错误时微信会返回错误码等信息，JSON数据包示例如下（该示例为AppID无效错误）：

```
{"errcode":40013,"errmsg":"invalid appid"}
```

4.1.3　实现代码

获取Access Token的实例代码如下所示：

```
$appid = "wxbad0b4x543aa0b5e";
$appsecret = "ed222a84da15cd24c4bdfa5d9adbabf2";
$url = "https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=$appid&secret=$appsecret";
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($ch);
curl_close($ch);
$jsoninfo = json_decode($output, true);
$access_token = $jsoninfo["access_token"];
```

上述代码执行以后，返回结果如下：

```
{"access_token":"NU7Kr6v9L9TQaqm5NE3OTPctTZx797Wxw4Snd2WL2HHBqLCiXlDVOw2l-Se0I-WmOLLniAYLAwzhbYhXNjbLc_KAA092cxkmpj5FpuqNO0IL7bB0Exz5s5qC9Umypy-rz2y441W9qgfnmNtIZWSjSQ","expire
```

这样，我们就获得了Access Token了。

4.2　自定义菜单

定义菜单能够帮助微信公众账号丰富界面，增强用户与微信公众账号互动，让用户更好、更快地理解微信公众账号所拥有的功能。

4.2.1　自定义菜单介绍

目前服务号和通过认证的订阅号均可申请自定义菜单。创建成功自定义菜单后，微信公众账号界面如图4-1所示。

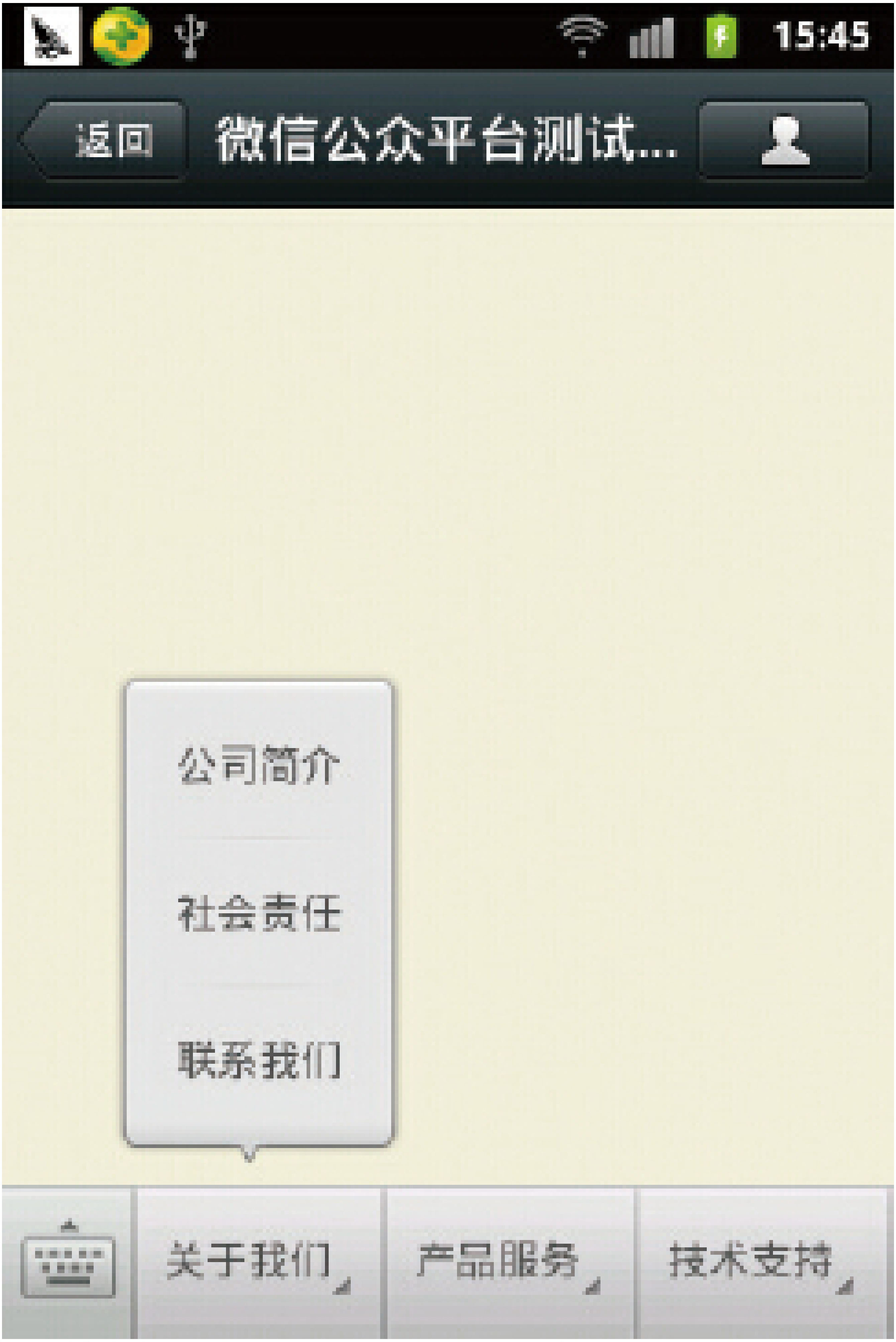


图4-1　自定义菜单效果

目前自定义菜单最多包括3个一级菜单，每个一级菜单最多包含5个二级菜单。一级菜单最多4个汉字，二级菜单最多7个汉字，多出来的部分将会以“http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/...”代替。请注意，创建自定义菜单后，由于微信客户端缓存，需要24小时微信客户端才会展现出来。可以重新关注微信公众账号，这样就能马上看到自定义菜单。

4.2.2 按钮类型

目前自定义菜单接口可实现以下几种类型按钮，具体如下所示。

1.click：单击推事件

用户单击click类型按钮后，微信服务器会通过消息接口推送消息类型为event的结构给开发者（参考消息接口指南），并且带上按钮中开发者填写的key值，开发者可以通过自定义的key值与用户进行交互。

2.view：跳转URL

用户单击view类型按钮后，微信客户端将会打开开发者在按钮中填写的网页URL，与网页授权获取用户基本信息接口结合，获得用户基本信息。

3.scancode_push：扫码推事件

用户单击按钮后，微信客户端将调用“扫一扫”功能，完成扫码操作后显示扫描结果（如果是URL，将进入URL），且会将扫码的结果传给开发者，开发者可以下发消息。

4.scancode_waitmsg：扫码推事件且弹出“消息接收中”提示框

用户单击按钮后，微信客户端将调用“扫一扫”功能，完成扫码操作后，将扫码的结果传给开发者，同时收起“扫一扫”功能，然后弹出“消息接收中”提示框，随后可能会收到开发者下发的消息。

5.pic_sysphoto：弹出系统拍照发图

用户单击按钮后，微信客户端将调用系统相机，完成拍照操作后，会将拍摄的相片发送给开发者，并推送事件给开发者，同时收起系统相机，随后可能会收到开发者下发的消息。

6.pic_photo_or_album：弹出拍照或者相册发图

用户单击按钮后，微信客户端将弹出选择器供用户选择“拍照”或者“从手机相册选择”。用户选择后即按照相应流程继续。

7.pic_weixin：弹出微信相册发图器

用户单击按钮后，微信客户端将调用微信相册，完成选择操作后，将选择的相片发送给开发者的服务器，并推送事件给开发者，同时收起相册，随后可能会收到开发者下发的消息。

8.location_select：弹出地理位置选择器

用户单击按钮后，微信客户端将调用地理位置选择功能，完成选择操作后，将选择的地理位置发送给开发者的服务器，同时收起位置选择功能，随后可能会收到开发者下发的消息。

4.2.3 创建菜单

创建菜单的接口如下所示：

```
https://api.weixin.qq.com/cgi-bin/menu/create?access_token=ACCESS_TOKEN
```

创建菜单时，需要将菜单内容组织成如下结构，以POST方式向微信服务器提交。

```
{
    "button": [
        {
            "name": "扫码",
            "sub_button": [
                {
                    "type": "scancode_waitmsg",
                    "name": "扫码带提示",
                    "key": "rselfmenu_0_0"
                },
                {
                    "type": "scancode_push",
                    "name": "扫码推事件",
                    "key": "rselfmenu_0_1"
                }
            ]
        },
        {
            "name": "发图",
            "sub_button": [
                {
                    "type": "pic_sysphoto",
                    "name": "系统拍照发图",
                    "key": "rselfmenu_1_0"
                },
                {
                    "type": "pic_photo_or_album",
                    "name": "拍照或者相册发图",
                    "key": "rselfmenu_1_1"
                },
                {
                    "type": "pic_weixin",
                    "name": "微信相册发图",
                    "key": "rselfmenu_1_2"
                }
            ]
        },
        {
            "name": "其他",
            "sub_button": [
                {
                    "name": "发送位置",
                    "type": "location_select",
                    "key": "rselfmenu_2_0"
                },
                {
                    "type": "click",
```

```
                "name": "今日歌曲",
                "key": "V1001_TODAY_MUSIC"
            },
            {
                "type": "view",
                "name": "搜索",
                "url": "http://www.soso.com/"
            }
        ]
    }
  ]
}
```

自定义菜单内容的参数说明如表4-3所示。

表4-3　自定义菜单参数说明

参数	是否必须	说明
button	是	一级菜单数组，个数应为 1 ～ 3 个
sub_button	否	二级菜单数组，个数应为 1 ～ 5 个
type	是	菜单的响应动作类型，目前有 click、view 两种类型
name	是	菜单标题，不超过 16 个字节，子菜单不超过 40 个字节
key	click 类型必须	菜单 KEY 值，用于消息接口推送，不超过 128 字节
url	view 类型必须	网页链接，用户单击菜单可打开链接，不超过 256 字节

给微信公众账号创建自定义菜单的代码实现如下所示：

```
$jsonmenu = '{
    "button": [
        {
            "name": "扫码",
            "sub_button": [
                {
                    "type": "scancode_waitmsg",
                    "name": "扫码带提示",
                    "key": "rselfmenu_0_0"
                },
                {
                    "type": "scancode_push",
                    "name": "扫码推事件",
                    "key": "rselfmenu_0_1"
                }
            ]
        },
        {
            "name": "发图",
            "sub_button": [
                {
                    "type": "pic_sysphoto",
                    "name": "系统拍照发图",
                    "key": "rselfmenu_1_0"
                },
                {
                    "type": "pic_photo_or_album",
                    "name": "拍照或者相册发图",
                    "key": "rselfmenu_1_1"
                },
                {
                    "type": "pic_weixin",
                    "name": "微信相册发图",
                    "key": "rselfmenu_1_2"
                }
            ]
        },
        {
            "name": "其他",
            "sub_button": [
                {
                    "name": "发送位置",
                    "type": "location_select",
                    "key": "rselfmenu_2_0"
                },
                {
                    "type": "click",
                    "name": "今日歌曲",
                    "key": "V1001_TODAY_MUSIC"
                },
                {
                    "type": "view",
                    "name": "搜索",
                    "url": "http://www.soso.com/"
                }
            ]
        }
    ]
}
';
$url = "https://api.weixin.qq.com/cgi-bin/menu/create?access_token=".$access_token;
$result = https_request($url, $jsonmenu);
var_dump($result);
function https_request($url,$data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

正确时返回JSON数据包如下：

```
{"errcode":0,"errmsg":"ok"}
```

错误时返回JSON数据包如下（示例为无效菜单名长度）：

```
{"errcode":40018,"errmsg":"invalid button name size"}
```


errcode为全局返回码，关于全局返回码的说明可以参考附录。

执行上述代码后，微信公众账号底部将出现菜单，效果如图4-1所示。

4.2.4 查询菜单

使用接口创建自定义菜单后，开发者还可使用接口查询自定义菜单的结构。

查询菜单的接口如下：

```
https://api.weixin.qq.com/cgi-bin/menu/get?access_token=ACCESS_TOKEN
```

查询自定义菜单代码实现如下所示：

```
$url = "https://api.weixin.qq.com/cgi-bin/menu/get?access_token=".$access_token;
$result = https_request($url);
var_dump($result);
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

正确时返回JSON数据包如下：


```
{"menu":{"button":[{"name":"扫码","sub_button":[{"type":"scancode_waitmsg","name":"扫码带提示","key":"rselfmenu_0_0","sub_button":[]},{"type":"scancode_push","name":"扫码推事件","
"sub_button":[]}]}]}}
```


4.2.5 删除菜单

使用接口创建自定义菜单后，开发者还可使用接口删除当前使用的自定义菜单。

删除菜单的接口如下：

```
https://api.weixin.qq.com/cgi-bin/menu/delete?access_token=ACCESS_TOKEN
```

查询自定义菜单代码实现如下所示：

```
$url = "https://api.weixin.qq.com/cgi-bin/menu/delete?access_token=".$access_token;
$result = https_request($url);
var_dump($result);
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

正确时返回JSON数据包如下：


```
{"errcode":0,"errmsg":"ok"}
```


4.2.6 菜单事件推送

用户单击自定义菜单后，如果菜单按钮类型为click，则微信公众平台会把此次单击事件推送给接口程序；按钮类型为view的菜单单击后不会上报，而是直接跳转到对应的URL。

单击click类型的菜单时，接口程序收到的XML数据包如下所示：


```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1392297442</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[CLICK]]></Event>
    <EventKey><![CDATA[公司简介]]></EventKey>
</xml>
```


单击scancode_waitmsg类型的菜单时，接口程序收到的XML数据包如下所示：

```
<xml>
        <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
        <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
        <CreateTime>1412075435</CreateTime>
        <MsgType><![CDATA[event]]></MsgType>
        <Event><![CDATA[scancode_waitmsg]]></Event>
        <EventKey><![CDATA[rselfmenu_0_0]]></EventKey>
        <ScanCodeInfo>
                <ScanType><![CDATA[qrcode]]></ScanType>
                <ScanResult><![CDATA[http://weixin.qq.com/r/pUNnf4HEX9wgrcUc9xa3]]></ScanResult>
                <EventKey><![CDATA[rselfmenu_0_0]]></EventKey>
        </ScanCodeInfo>
</xml>
```

单击scancode_push类型的菜单时，接口程序收到的XML数据包如下所示，微信会返回该XML给接口程序，但用户收不到这个消息，而是直接运行解码后的内容，比如直接进入关注界面：

```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075451</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[scancode_push]]></Event>
    <EventKey><![CDATA[rselfmenu_0_1]]></EventKey>
    <ScanCodeInfo>
        <ScanType><![CDATA[qrcode]]></ScanType>
        <ScanResult><![CDATA[http://weixin.qq.com/r/pUNnf4HEX9wgrcUc9xa3]]></ScanResult>
        <EventKey><![CDATA[rselfmenu_0_1]]></EventKey>
    </ScanCodeInfo>
</xml>
```

单击pic_weixin类型的菜单时，微信客户端将调用系统相机，用户可以选择已有相片或者进行拍照，微信会将相片发送给开发者。下面是一次选择3张相片时的XML数据：

```
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075552</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[pic_weixin]]></Event>
    <EventKey><![CDATA[rselfmenu_1_2]]></EventKey>
    <SendPicsInfo>
        <Count>3</Count>
        <PicList>
            <item>
                <PicMd5Sum><![CDATA[a6ab74b73a298f49f6fc66f41f88d3c7]]></PicMd5Sum>
            </item>
            <item>
                <PicMd5Sum><![CDATA[5b9aea2b3683c5c21aaf3629739cafd9]]></PicMd5Sum>
            </item>
            <item>
                <PicMd5Sum><![CDATA[2325ff3f39705ac987d3e0660906791d]]></PicMd5Sum>
            </item>
        </PicList>
        <EventKey><![CDATA[rselfmenu_1_2]]></EventKey>
    </SendPicsInfo>
</xml>
```

上述消息XML数据下发后，微信会继续推送3个图片消息给接口：

```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075562</CreateTime>
    <MsgType><![CDATA[image]]></MsgType>
<PicUrl><![CDATA[http://mmbiz.qpic.cn/mmbiz/qOTIrRtMKFxZQo3cQ6vYxmfteo0fw2Y00MuuFS6OjlNvTkx0nBQJibjjafQMf4dCtJwyMeP5e3SQCuJpcoJdb8Q/0]]></PicUrl>
    <MsgId>6064818358471506877</MsgId>
    <MediaId><![CDATA[AM8Gbiu7S2vTIMS2l9zNCWiMCpeM8wccmKdFxmstHiuAybCbB-08dTQJN40cCIBD]]></MediaId>
</xml>
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075564</CreateTime>
    <MsgType><![CDATA[image]]></MsgType>
<PicUrl><![CDATA[http://mmbiz.qpic.cn/mmbiz/qOTIrRtMKFxZQo3cQ6vYxmfteo0fw2Y0ellMniarWto4zL0pWKhr6a0FSARDeXprSgd0JmgU0YQo2iagGsYc1iaLA/0]]></PicUrl>
    <MsgId>6064818367061441471</MsgId>
    <MediaId><![CDATA[2CJKZ2mft-5hWEkVcVmnKwyvp3y59Uzv7YdpBsnGIyTlkMNb0ytU4oimTgimBGyP]]></MediaId>
</xml>
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075567</CreateTime>
    <MsgType><![CDATA[image]]></MsgType>
<PicUrl><![CDATA[http://mmbiz.qpic.cn/mmbiz/qOTIrRtMKFxZQo3cQ6vYxmfteo0fw2Y0mwQoENttnmYAz0VTicY3xkO0lhn9mE5r3FB4cF04kia50riajcjY1qvzA/0]]></PicUrl>
    <MsgId>6064818379946343361</MsgId>
    <MediaId><![CDATA[VICm1-y8w_MmdpcLYWP0u12HVXrWdhm3c3tdq1PTsbiFFgH6YOcmmIEbZ27z-Qcl]]></MediaId>
</xml>
```

对于上述每个图片的消息，后台收到都能进行单独回应，用户可以收到响应消息。

单击pic_photo_or_album类型的菜单之后，和pic_weixin一样，先推送菜单事件，再推送图片消息：

```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075614</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[pic_photo_or_album]]></Event>
    <EventKey><![CDATA[rselfmenu_1_1]]></EventKey>
    <SendPicsInfo>
        <Count>1</Count>
        <PicList>
            <item>
                <PicMd5Sum><![CDATA[58f4989164b2a432889e64fa81632152]]></PicMd5Sum>
            </item>
        </PicList>
        <EventKey><![CDATA[rselfmenu_1_1]]></EventKey>
    </SendPicsInfo>
</xml>
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075618</CreateTime>
    <MsgType><![CDATA[image]]></MsgType>
    <PicUrl><![CDATA[http://mmbiz.qpic.cn/mmbiz/qOTIrRtMKFxZQo3cQ6vYxmfteo0fw2Y0fwSdF54rribCj9zqDG0wVDsiasKuUC8gskbuZmX43a1hs9esySIvmrqA/0]]></PicUrl>
    <MsgId>6064818598989675467</MsgId>
    <MediaId><![CDATA[Jh7Bsjaodmz-Dc_Hdz_wRzY3RiFxNvA4JHpUWRNRCIgps19ANsDFIOYOUV1dIVYV]]></MediaId>
</xml>
```

单击pic_sysphoto类型的菜单后没有XML数据推送过来，它调用的是手机里面的照相机功能，而拍照以后再发过来时就收到一个图片消息，XML数据如下所示。

```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
```

```
    <CreateTime>1412135923</CreateTime>
    <MsgType><![CDATA[image]]></MsgType>
    <PicUrl><![CDATA[http://mmbiz.qpic.cn/mmbiz/qOTIrRtMKFxGzYCchrNiczZ9aCd1fgCEWbicHGvmaz1d7VZOSMqoLB4RL69rK2Fos6fSM4FJRkLtiajpe32ibkicw1A/0]]></PicUrl>
    <MsgId>6065077606992462276</MsgId>
    <MediaId><![CDATA[PqRRUPLhMc27EG5F4cCtTozBsin7SY8kPRRuAF-CwPl5MzR5ewE5oji0FhGTDl6_]]> </MediaId>
</xml>
```

单击location_select类型的菜单之后，将会调用发送位置功能，在用户发送位置之后，会再推送一个地理位置消息功能给用户，两个消息的XML数据分别如下所示。

```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075681</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[location_select]]></Event>
    <EventKey><![CDATA[rselfmenu_2_0]]></EventKey>
    <SendLocationInfo>
        <Location_X><![CDATA[22.538145]]></Location_X>
        <Location_Y><![CDATA[113.952298]]></Location_Y>
        <Scale><![CDATA[13]]></Scale>
        <Label><![CDATA[深圳市南山区高新南一道9号]]></Label>
        <Poiname><![CDATA[]]></Poiname>
        <EventKey><![CDATA[rselfmenu_2_0]]></EventKey>
    </SendLocationInfo>
</xml>
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdHniQs]]></FromUserName>
    <CreateTime>1412075681</CreateTime>
    <MsgType><![CDATA[location]]></MsgType>
    <Location_X>22.538145</Location_X>
    <Location_Y>113.952298</Location_Y>
    <Scale>13</Scale>
    <Label><![CDATA[深圳市南山区高新南一道9号飞亚达科技大厦方倍工作室]]></Label>
    <MsgId>6064818869572615123</MsgId>
</xml>
```

上述菜单的XML数据包相关参数说明如表4-4所示。

表4-4 自定义菜单推送参数说明

参数	说明
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	消息类型，event
Event	事件类型
EventKey	事件 KEY 值，与自定义菜单接口中 KEY 值对应

消息接口中，响应自定义菜单单击事件的核心代码如下所示：

```
//接收事件消息
private function receiveEvent($object)
{
    $content = "";
    switch ($object->Event)
    {
        case "subscribe":
            $content = "欢迎关注方倍工作室 ";
            $content .= (!empty($object->EventKey))?("\n来自二维码场景 ".
str_replace("qrscene_","",$object->EventKey)):"";
            break;
        case "unsubscribe":
            $content = "取消关注";
            break;
        case "CLICK":
            switch ($object->EventKey)
            {
                case "COMPANY":
                    $content = array();
               $content[] = array("Title"=>"方倍工作室", "Description"=>"", "PicUrl"=>"http://discuz.comli.com/weixin/weather/icon/cartoon.jpg", "Url" =>"http://m.cnblogs.com/?u
                    break;
                default:
                    $content = "单击菜单："..$object->EventKey;
                    break;
            }
            break;
        case "VIEW":
            $content = "跳转链接 ".$object->EventKey;
            break;
        case "SCAN":
            $content = "扫描场景 ".$object->EventKey;
            break;
        case "LOCATION":
            $content = "上传位置：纬度 ".$object->Latitude.";经度 ".$object->Longitude;
            break;
        case "scancode_waitmsg":
            $content = "扫码带提示：类型 ".$object->ScanCodeInfo->ScanType." 结果：".$object->ScanCodeInfo->ScanResult;
            break;
        case "scancode_push":
            $content = "扫码推事件";
            break;
        case "pic_sysphoto":
            $content = "系统拍照";
            break;
        case "pic_weixin":
            $content = "相册发图：数量 ".$object->SendPicsInfo->Count;
            break;
        case "pic_photo_or_album":
            $content = "拍照或者相册：数量 ".$object->SendPicsInfo->Count;
            break;
        case "location_select":
            $content = "发送位置：标签 ".$object->SendLocationInfo->Label;
            break;
        default:
            $content = "receive a new event: ".$object->Event." \n技术支持 方倍工作室";
            break;
    }
    if(is_array($content)){
        if (isset($content[0]['PicUrl'])){
            $result = $this->transmitNews($object, $content);
        }else if (isset($content['MusicUrl'])){
            $result = $this->transmitMusic($object, $content);
        }
    }else{
        $result = $this->transmitText($object, $content);
    }
```

```
    return $result;
}
```

4.3 语音识别

4.3.1 接收识别结果

开通语音识别功能以后，用户每次发送语音给微信公众账号时，微信会在推送的语音消息XML数据包中增加一个Recongnition字段。该字段为语音识别出的文本内容。

用户发送语音如图4-2所示。

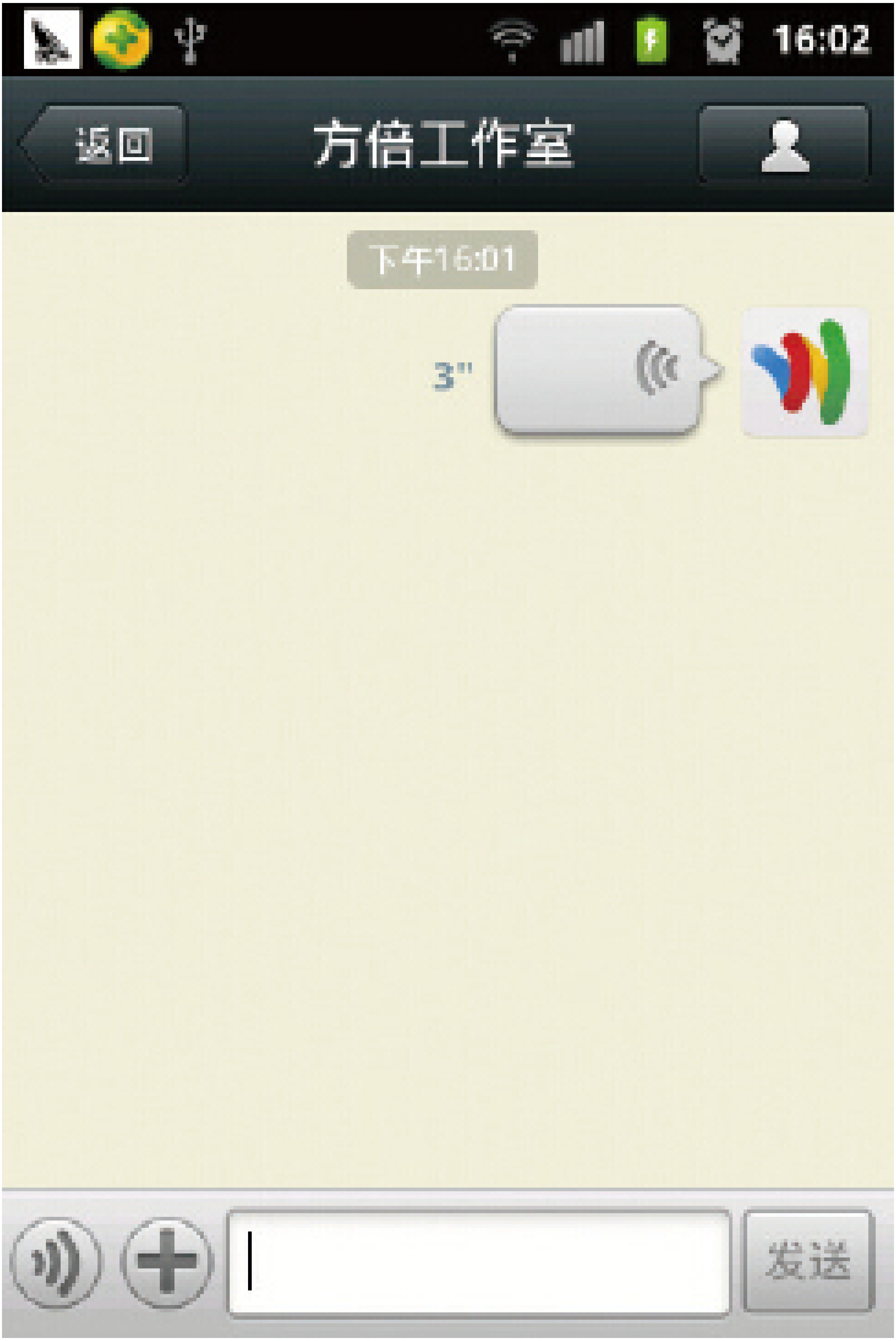

16:02
返回
方倍工作室
下午16:01
3"
发送

图4-2　用户发送语音

用户发送语音消息时，XML数据包如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdH
niQs]]></FromUserName>
    <CreateTime>1392298316</CreateTime>
    <MsgType><![CDATA[voice]]></MsgType>
    <MediaId><![CDATA[FEATTV4nFtsvAuCKqYk11V0iLJMqy7JSUjRws4CSeth5cwrw6dgW0zPFQcmoV9Vy]]></MediaId>
    <Format><![CDATA[amr]]></Format>
    <MsgId>5979875733695893107</MsgId>
    <Recognition><![CDATA[深圳天气怎么样]]></Recognition>
</xml>
```

语音消息参数说明如表4-5所示。

表4-5　语音消息参数说明

参数	说明
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	语音类型，voice
MediaID	语音消息媒体 id，可以调用多媒体文件下载接口拉取该媒体
Format	语音格式：amr
Recognition	语音识别结果，UTF8 编码
MsgID	消息 id，64 位整型

使用语音识别接口可以简化输入，让用户从繁琐的键盘输入中解放出来，对于一些简单内容的输入，用户可以直接使用语音与微信公众账号进行交互。

4.3　语音识别

4.3.1　接收识别结果

开通语音识别功能以后，用户每次发送语音给微信公众账号时，微信会在推送的语音消息XML数据包中增加一个Recongnition字段。该字段为语音识别出的文本内容。

用户发送语音如图4-2所示。

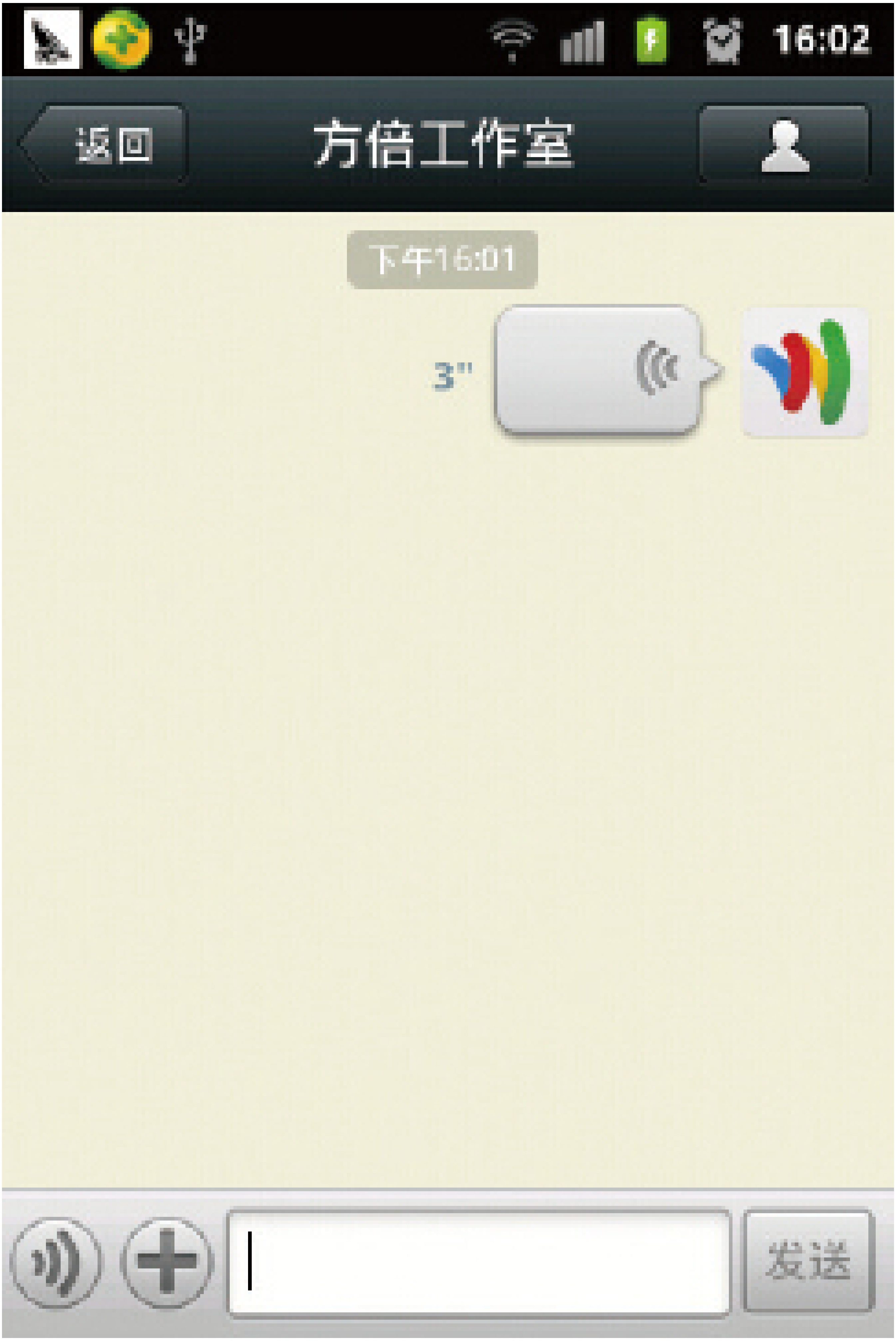

16:02
返回
方倍工作室
下午16:01
3"
发送

图4-2　用户发送语音

用户发送语音消息时，XML数据包如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_82479813ed64]]></ToUserName>
    <FromUserName><![CDATA[ojpX_jig-gyi3_Q9fHXQ4rdH
niQs]]></FromUserName>
    <CreateTime>1392298316</CreateTime>
    <MsgType><![CDATA[voice]]></MsgType>
    <MediaId><![CDATA[FEATTV4nFtsvAuCKqYk11V0iLJMqy7JSUjRws4CSeth5cwrw6dgW0zPFQcmoV9Vy]]></MediaId>
    <Format><![CDATA[amr]]></Format>
    <MsgId>5979875733695893107</MsgId>
    <Recognition><![CDATA[深圳天气怎么样]]></Recognition>
</xml>
```

语音消息参数说明如表4-5所示。

表4-5　语音消息参数说明

参数	说明
ToUserName	接收方微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	语音类型，voice
MediaID	语音消息媒体 id，可以调用多媒体文件下载接口拉取该媒体
Format	语音格式：amr
Recognition	语音识别结果，UTF8 编码
MsgID	消息 id，64 位整型

使用语音识别接口可以简化输入，让用户从繁琐的键盘输入中解放出来，对于一些简单内容的输入，用户可以直接使用语音与微信公众账号进行交互。

4.3.2　处理识别消息

微信公众账号接收到语音消息后，由于数据包中已经包含了识别结果的文字内容，所以原则上，应当将其当作文本消息一样处理，代码如下所示：

```
public function responseMsg()
{
    $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
    if (!empty($postStr)){
        $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
        $RX_TYPE = trim($postObj->MsgType);
        switch ($RX_TYPE)
        {
            case "event":
                $result = $this->receiveEvent($postObj);
                break;
            case "text":
            case "voice":
                $result = $this->receiveText($postObj);
                break;
        }
        echo $result;
    }else {
        echo "";
        exit;
    }
}
```

而在接收函数中，再将语音消息的识别结果或文本消息的文本赋给同一个变量。相关代码如下：

```
private function receiveText($object)
{
    if (isset($object->Recognition)){
        $keyword = trim($object->Recognition);
        $mediaid = trim($object->MediaID);
    }else{
        $keyword = trim($object->Content);
    }
    //To Do List
}
```

需要说明的是，受个体口音、重音、音调、音量和发音速度、环境噪音以及语义理解等诸多因素的影响，语音识别目前很难做到百分之百的准确率。随着人们地不断研究及探索，语音识别技术也在不断地发展进步。

4.4　客服接口

当用户主动发消息给微信公众账号的时候（包括发送信息、单击自定义菜单click事件、订阅事件、扫描二维码事件、支付成功事件、用户维权），微信将会把消息数据推送给开发者。开发者在一段时间内（目前为48小时）可以调用客服消息接口，通过POST一个JSON数据包来发送消息给普通用户，在48小时内不限制发送次数。此接口主要用于客服等有人工消息处理环节的功能，方便开发者为用户提供更加优质的服务。

4.4.1 客服接口实现

我们可以直接构造文本和图文消息回复给用户。而音乐、图片、语音、视频四种消息，需要首先获得素材的media_id，这可以从用户发送过来的消息那获得，也可以通过上传多媒体文件来获得。

发送客服消息的接口如下：

```
https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=ACCESS_TOKEN
```

发送各种类型的客服消息介绍如下：

1.向用户发送文本消息

向用户发送文本消息时，数据格式如下：

```
{
    "touser":"OPENID",
    "msgtype":"text",
    "text":
    {
         "content":"Hello World"
    }
}
```

该数据参数说明如表4-6所示。

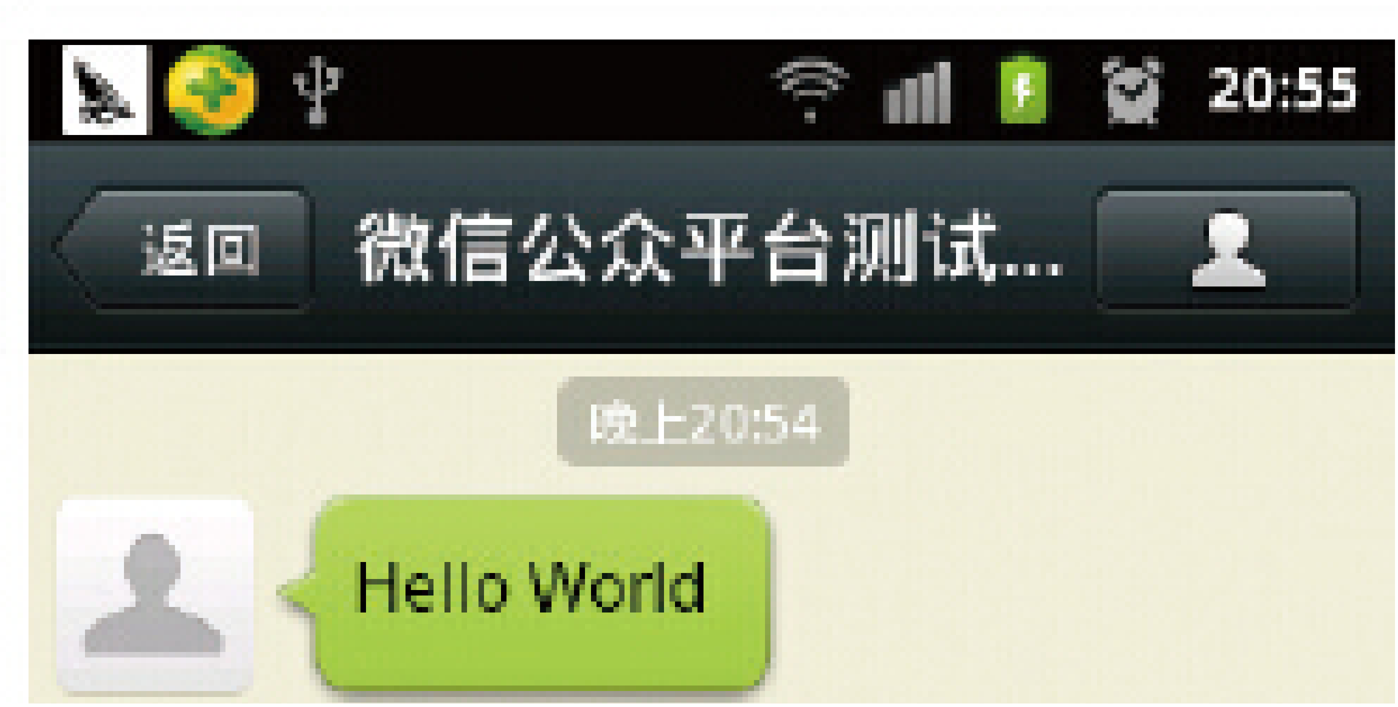
表4-6 客服接口文本消息参数说明

参数	是否必须	说明	参数	是否必须	说明
access_token	是	调用接口凭证	msgtype	是	消息类型，text
touser	是	普通用户 OpenID	content	是	文本消息内容

向用户发送文本消息的代码实现如下：

```
$access_token = "nFX6GFsspSLBKJLgMQ3kj1YM8_FchRE7vE2ZOIlmfiCOQntZKnBwuOen2GCBpFHBYS4QLGX9fGoVfA36tftME2sRiYsKPzgGQKU-ygU7x8cgy_1tlQ4n1mhSumwQEGy6PK6rdTdo8O8GROuGE3Hiag";
$openid = "o7Lp5t6n59DeX3U0C7Kric9qEx-Q";
$data = '{
    "touser":"'.$openid.'",
    "msgtype":"text",
    "text":
    {
         "content":"Hello World"
    }
}';
$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$access_token;
$result = https_request($url,$data);
var_dump($result);
function https_request($url,$data)
{
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($curl, CURLOPT_POST, 1);
    curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$result = curl_exec($curl);
if (curl_errno($curl)) {
return 'Errno'.curl_error($curl);
    }
    curl_close($curl);
return$result;
}
```

上述代码执行后，效果如图4-3所示。



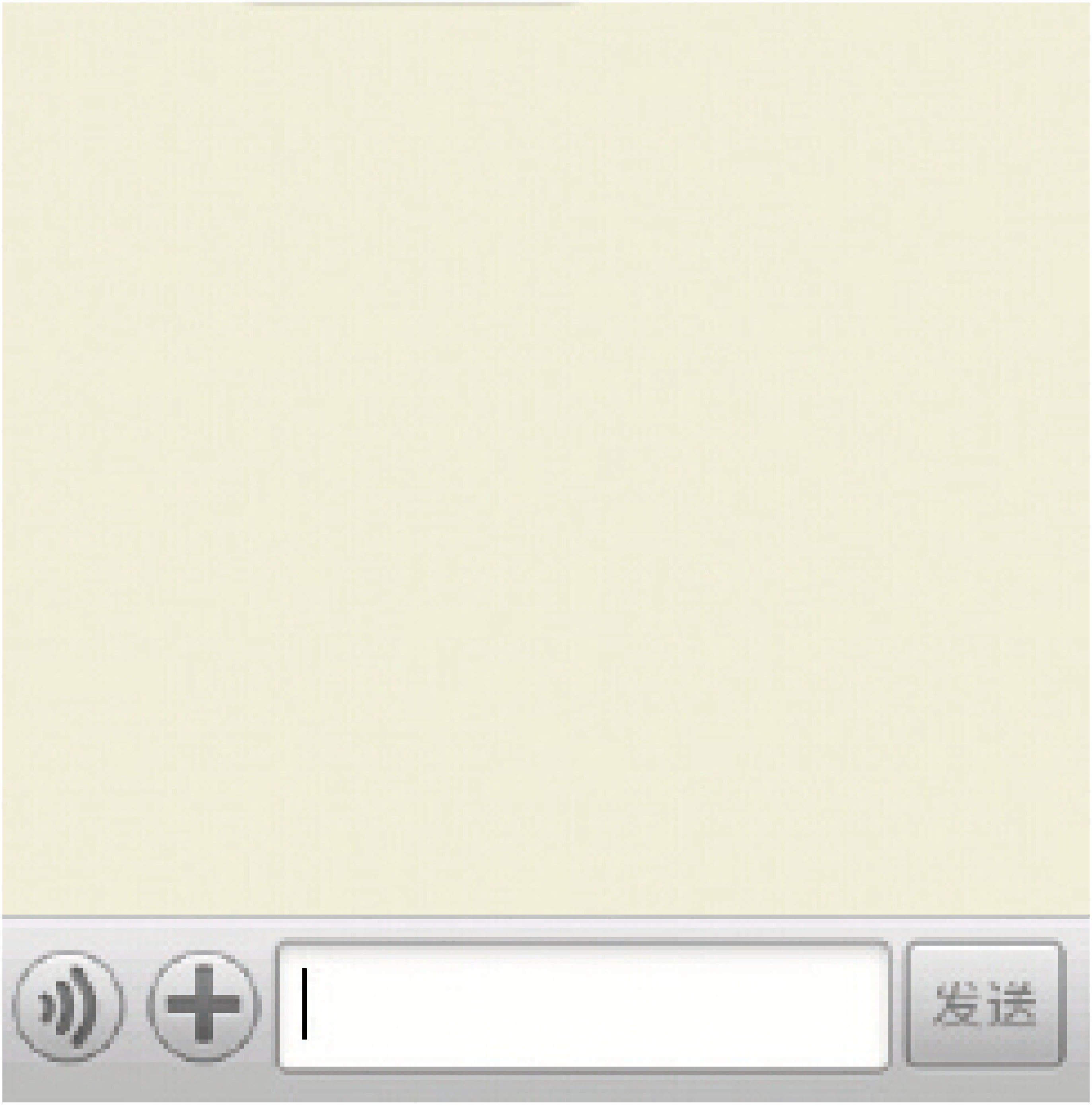


图4-3　微信公众账号发送文本消息

2.向用户发送图文消息

向用户发送图文消息时，数据格式如下：

```
{
    "touser":"OPENID",
    "msgtype":"news",
    "news":{
        "articles": [
         {
             "title":"Happy Day",
             "description":"Is Really A Happy Day",
             "url":"URL",
             "picurl":"PIC_URL"
         },
         {
             "title":"Happy Day",
             "description":"Is Really A Happy Day",
             "url":"URL",
             "picurl":"PIC_URL"
         }
         ]
    }
}
```

该数据参数说明如表4-7所示。

表4-7　客服接口图文消息参数说明

参数	是否必须	说明
access_token	是	调用接口凭证
touser	是	普通用户 OpenID
msgtype	是	消息类型，news
title	否	标题
description	否	描述
url	否	单击后跳转的链接
picurl	否	图文消息的图片链接，支持 JPG、PNG 格式，较好的效果为大图 640×320 像素，小图 80×80 像素

向用户发送图文消息的代码实现如下：

```
$access_token = "NU7Kr6v9L9TQaqm5NE3OTPctTZx797Wxw4Snd2WL2HHBqLCiXlDVOw2l-Se0I-WmOLLniAYLAwzhbYhXNjbLc_KAA092cxkmpj5FpuqNO0IL7bB0Exz5s5qC9Umypy-rz2y441W9qgfnmNtIZWSjSQ";
$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$access_token;
$openid = "oLVPpjqs9BhvzwPj5A-vTYAX3GLc";
//发送图文
$data = '{
    "touser": "'.$openid.'",
    "msgtype": "news",
    "news": {
        "articles": [
            {
                "title": "深圳天气预报",
                "description": "",
                "picurl": "",
                "url": ""
            },
            {
                "title": "12月04日周三 13℃~23℃多云微风",
                "description": "",
                "picurl": "http://discuz.comli.com/weixin/weather/d01.jpg",
                "url": ""
            },
            {
                "title": "12月05日周四 16℃~24℃晴微风",
                "description": "",
                "picurl": "http://discuz.comli.com/weixin/weather/d00.jpg",
                "url": ""
            },
            {
                "title": "12月06日周五 16℃~24℃多云微风",
                "description": "",
                "picurl": "http://discuz.comli.com/weixin/weather/d01.jpg",
                "url": ""
            }
        ]
    }
}';
$result = https_request($url,$data);
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

上述代码执行后，效果如图4-4所示。

21:03
返回
微信公众平台测试...
晚上21:03
深圳天气预报
12月04日 周三 13℃-23℃ 多云 微风
12月05日 周四 16℃-24℃ 晴 微风
12月06日 周五 16℃-24℃ 多云 微风
发送

图4-4 微信公众账号发送图文消息

3.向用户发送音乐消息

向用户发送音乐消息时，数据格式如下：

```
{
    "touser":"OPENID",
    "msgtype":"music",
    "music":
    {
      "title":"MUSIC_TITLE",
      "description":"MUSIC_DESCRIPTION",
      "musicurl":"MUSIC_URL",
      "hqmusicurl":"HQ_MUSIC_URL",
      "thumb_media_id":"THUMB_MEDIA_ID"
    }
}
```

该数据参数说明如表4-8所示。

表4-8 客服接口音乐消息参数说明

参数	是否必须	说明
access_token	是	调用接口凭证
touser	是	普通用户 OpenID
msgtype	是	消息类型，music
title	否	音乐标题
description	否	音乐描述
musicurl	是	音乐链接
hqmusicurl	是	高品质音乐链接，Wi-Fi 环境优先使用该链接播放音乐
thumb_media_id	是	缩略图的媒体 id

发送音乐消息的代码实现如下：

```
$access_token = "NU7Kr6v9L9TQaqm5NE3OTPctTZx797Wxw4Snd2WL2HHBqLCiXlDVOw2l-Se0I-WmOLLniAYLAwzhbYhXNjbLc_KAA092cxkmpj5FpuqNO0IL7bB0Exz5s5qC9Umypy-rz2y441W9qgfnmNtIZWSjSQ";
$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$access_token;
$openid = "oLVPpjqs9BhvzwPj5A-vTYAX3GLc";
//发送音乐
$data = '{
    "touser": "'.$openid.'",
    "msgtype": "music",
    "music": {
        "title": "最炫民族风",
        "description": "凤凰传奇",
        "musicurl": "http://122.228.226.40/music/zxmzf.mp3",
        "hqmusicurl": "http://122.228.226.40/music/zxmzf.mp3",
        "thumb_media_id": "b8as-GpA_EFqVoPY7vPT3fpVZBWJb27K77De2dc_0FZml-UExlTMP7IVMz89uh3W"
    }
}';
$result = https_request($url,$data);
var_dump($result);
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

上述代码执行后，效果如图4-5所示。

21:27
返回
微信公众平台测试...
晚上21:27
最炫民族风
凤凰传奇
发送

图4-5　微信公众账号发送音乐消息

4.向用户发送图片消息

向用户发送图片消息时，数据格式如下：

```
{
    "touser":"OPENID",
    "msgtype":"image",
    "image":
    {
      "media_id":"MEDIA_ID"
    }
}
```

该数据参数说明如表4-9所示。

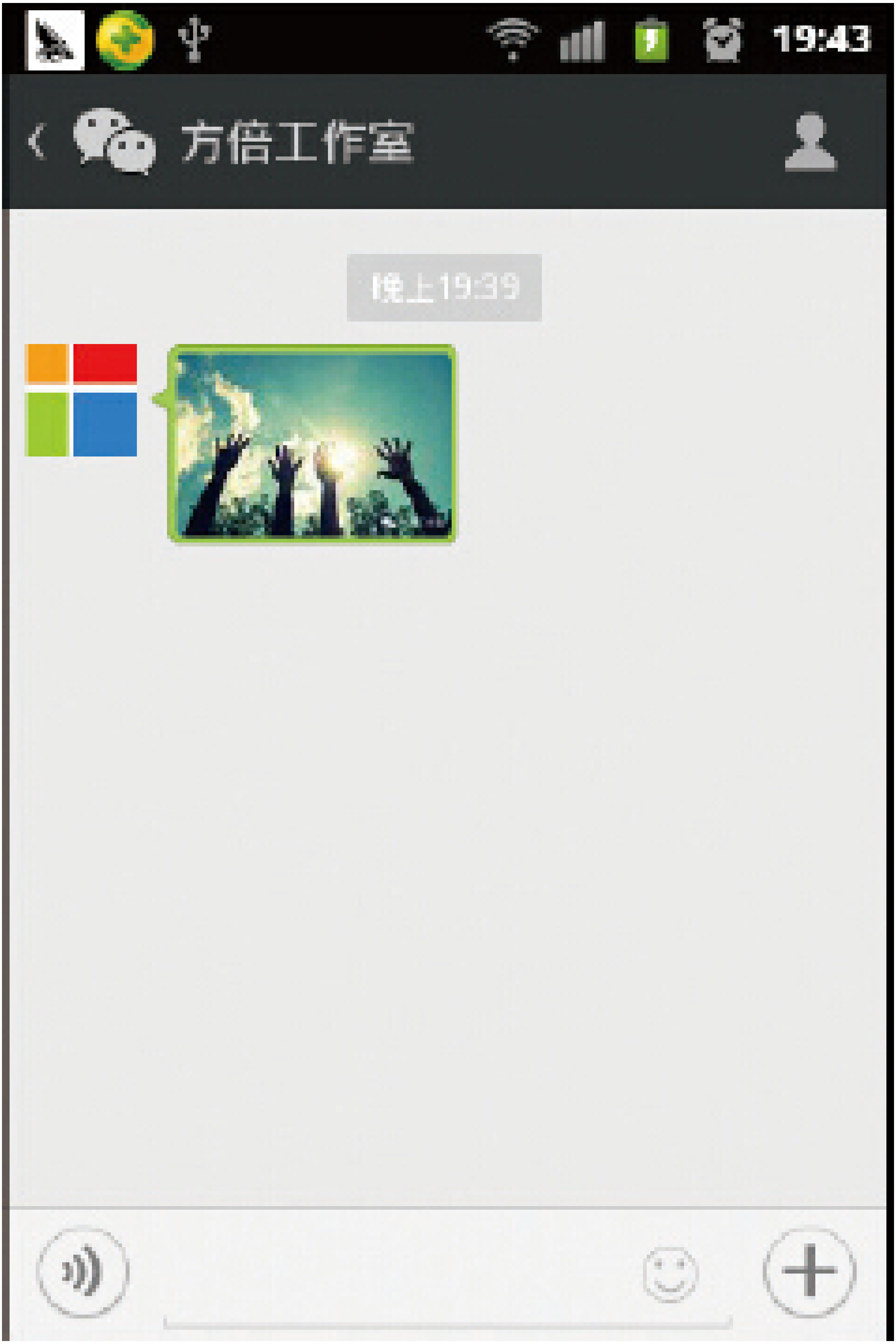
表4-9　客服接口图片消息参数说明

参数	是否必须	说明	参数	是否必须	说明
access_token	是	调用接口凭证	msgtype	是	消息类型，image
touser	是	普通用户 OpenID	media_id	是	发送的图片的媒体 id

发送图片消息的代码实现如下：

```
$access_token = "NU7Kr6v9L9TQaqm5NE3OTPctTZx797Wxw4Snd2WL2HHBqLCiXlDVOw2l-Se0I-WmOLLniAYLAwzhbYhXNjbLc_KAA092cxkmpj5FpuqNO0IL7bB0Exz5s5qC9Umypy-rz2y441W9qgfnmNtIZWSjSQ";
$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$access_token;
$openid = "oLVPpjqs9BhvzwPj5A-vTYAX3GLc";
//发送图片
$data = '{
    "touser":"'.$openid.'",
    "msgtype":"image",
    "image":
    {
      "media_id":"jViS8sQUXIh6rTdkz2qUOD5pvChYpp50i9rsLR4YnCm-pqdpiHhz6xbN2KAboScV"
    }
}';
$result = https_request($url,$data);
var_dump($result);
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

代码执行后，效果如图4-6所示。



19:43
方倍工作室
晚上19:39

图4-6　微信公众账号发送图片消息

5.向用户发送语音消息

向用户发送语音消息时，数据格式如下：

```
{
    "touser":"OPENID",
    "msgtype":"voice",
    "voice":
    {
      "media_id":"MEDIA_ID"
    }
}
```

该数据参数说明如表4-10所示。

表4-10　客服接口语音消息参数说明

参数	是否必须	说明	参数	是否必须	说明
access_token	是	调用接口凭证	msgtype	是	消息类型，voice
touser	是	普通用户 OpenID	media_id	是	发送的语音的媒体 id

向用户发送语音消息的代码实现如下：

```
$access_token = "NU7Kr6v9L9TQaqm5NE3OTPctTZx797Wxw4Snd2WL2HHBqLCiXlDVOw2l-Se0I-WmOLLniAYLAwzhbYhXNjbLc_KAA092cxkmpj5FpuqNO0IL7bB0Exz5s5qC9Umypy-rz2y441W9qgfnmNtIZWSjSQ";
$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$access_token;
$openid = "oLVPpjqs9BhvzwPj5A-vTYAX3GLc";
//发送语音
$data= '
{
    "touser": "'.$openid.'",
    "msgtype": "voice",
    "voice": {
        "media_id": "gAn8fV1hQB8mtLf6X2YTN-3ySaBMlzvbCLAzFUhPQDRVrWA1o9tRCnzxa4bEzUiM"
    }
}';
$result = https_request($url,$data);
var_dump($result);
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

上述代码执行后，效果如图4-7所示。

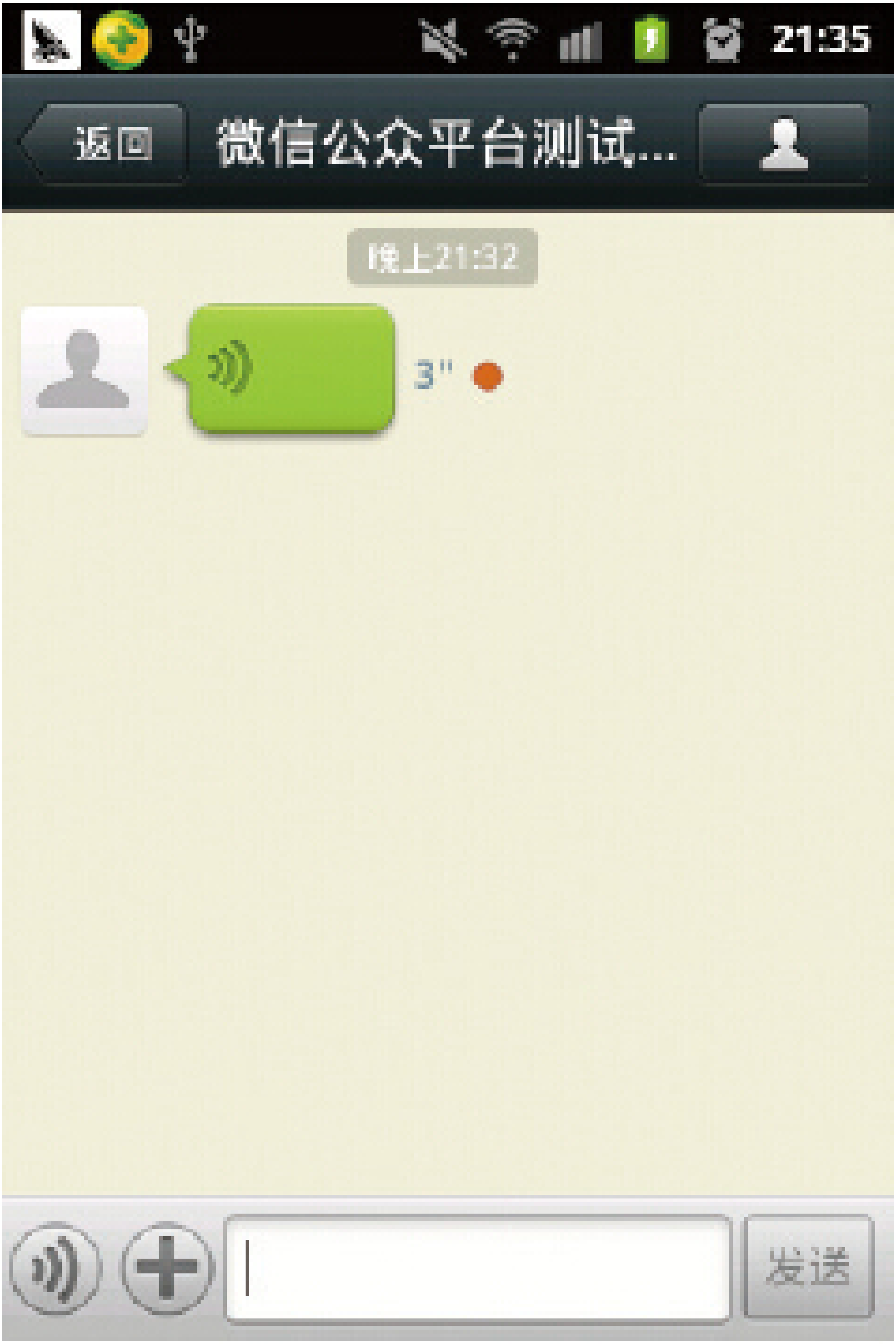


图4-7　微信公众账号发送语音消息

6.向用户发送视频消息

向用户发送视频消息时，数据格式如下：

```
{
    "touser":"OPENID",
    "msgtype":"video",
    "video":
    {
      "media_id":"MEDIA_ID",
      "title":"TITLE",
      "description":"DESCRIPTION"
    }
}
```

该数据参数说明如表4-11所示。

表4-11　客服接口视频消息参数说明

参数	是否必须	说明	参数	是否必须	说明
access_token	是	调用接口凭证	media_id	是	发送的视频的媒体 id
touser	是	普通用户 OpenID	thumb_media_id	是	发送的视频缩略图的媒体 id
msgtype	是	消息类型，video			

向用户发送视频消息的代码实现如下：

```
$access_token = "NU7Kr6v9L9TQaqm5NE3OTPctTZx797Wxw4Snd2WL2HHBqLCiXlDVOw2l-Se0I-WmOLLniAYLAwzhbYhXNjbLc_KAA092cxkmpj5FpuqNO0IL7bB0Exz5s5qC9Umypy-rz2y441W9qgfnmNtIZWSjSQ";
$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$access_token;
$openid = "oLVPpjqs9BhvzwPj5A-vTYAX3GLc";
//发送视频
$data = '{
    "touser": "'.$openid.'",
    "msgtype": "video",
    "video": {
        "media_id": "zbpy8wXj2UtIKB_56QUddCYLmWPLVeHoKqE94h2-nB9weSU65nVKTpE909Mvi8b5",
        "thumb_media_id":"b8as-GpA_EFqVoPY7vPT3fpVZBWJb27K77De2dc_0FZml-UExlTMP7IVMz
89uh3W"
    }
}';
$result = https_request($url,$data);
var_dump($result);
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

上述代码执行后，效果如图4-8所示。

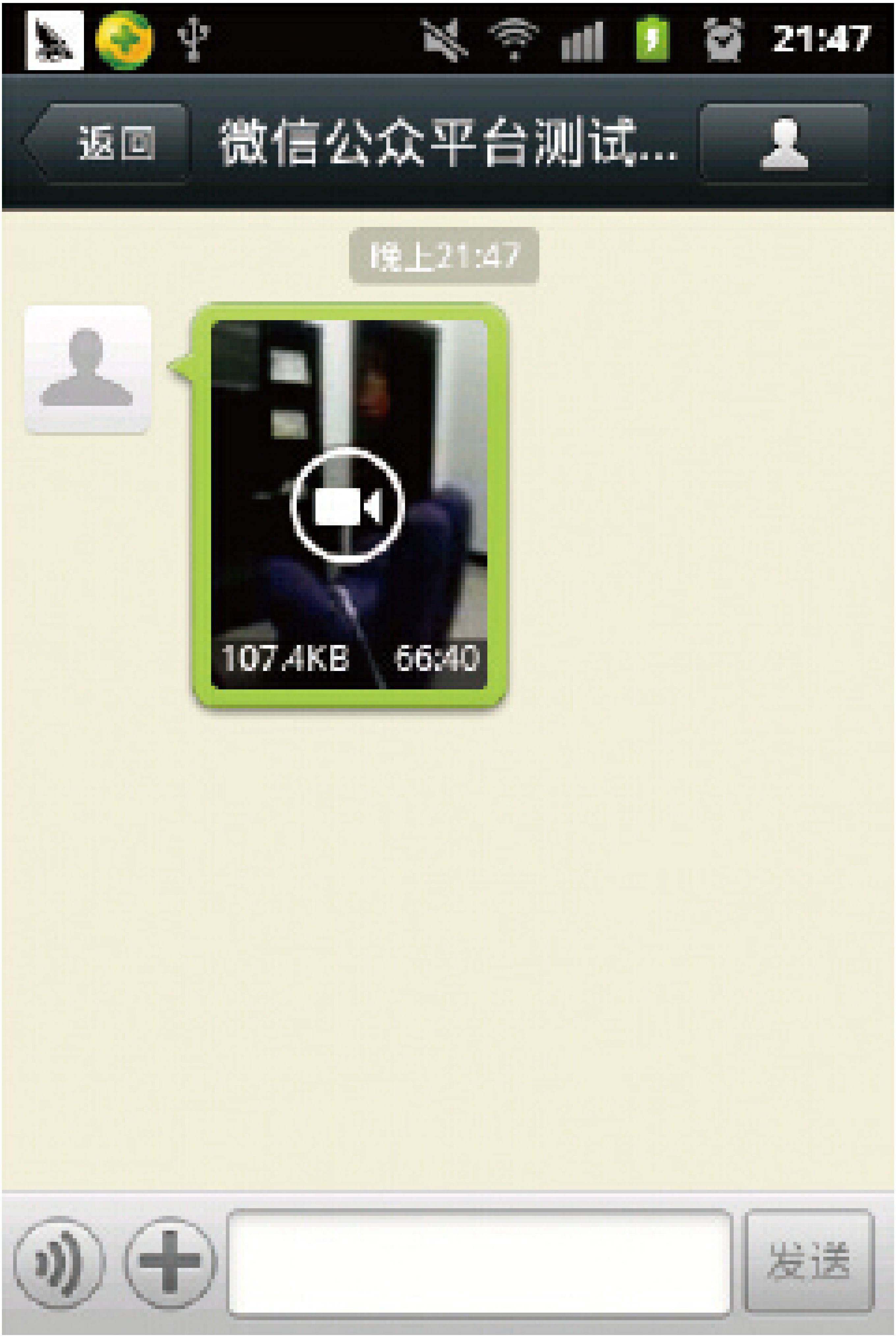

21:47
返回
微信公众平台测试...
晚上21:47
107.4KB
66:40
发送

图4-8　微信公众账号发送视频消息

4.4.2　消息接口与客服接口相结合

下面介绍使用消息接口与客服接口相结合的方式来回复用户。

首先，我们在消息接口中收到文本消息的时候，获取OpenID，相关代码如下：

```
//调用客服接口回复
$access_token = "nFX6GFsspSLBKJLgMQ3kj1YM8_FchRE7vE2ZOIlmfiCOQntZKnBwuOen2GCBpFHBYS4QLGX9fGoVfA36tftME2sRiYsKPzgGQKU-ygU7x8cgy_1tlQ4n1mhSumwQEGy6PK6rdTdo8O8GROuGE3Hiag";
$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$access_token;
$openid = $object->FromUserName;
```

然后向该OpenID发送客服消息，这里可以同时发送文本消息和音乐消息，相关代码如下：

```
//发送最炫民族风的介绍
$data = '{
    "touser":"'.$openid.'",
    "msgtype":"text",
    "text":
    {
         "content":"《最炫民族风》是凤凰传奇演唱的歌曲，是其第三张专辑《最炫民族风》的主打歌，于2009年5月27日全亚洲同步发行，2012年3月起在世界范围内走红。其彩铃下载量超过5000万。"
    }
}';
$this->https_request($url,$data);
//发送最炫民族风的音乐
$data = '{
    "touser": "'.$openid.'",
    "msgtype": "music",
    "music": {
        "title": "最炫民族风",
        "description": "凤凰传奇",
        "musicurl": "http://122.228.226.40/music/zxmzf.mp3",
        "hqmusicurl": "http://122.228.226.40/music/zxmzf.mp3",
        "thumb_media_id": "jVyS3KRGXvfeLHcnFVDSx07LeFhff-qwH0tVxRyU5RMBtC3aC14ta5HlGb6eK0_d"
    }
}';
$this->https_request($url,$data);
```

发送完客服消息之后，还可以继续使用消息接口回复，相关代码如下：

```
$contentStr = "以上是为您找到的关于 ".$object->Content." 的内容";
if (is_array($contentStr)){
$resultStr = $this->transmitNews($object, $contentStr);
}else{
$resultStr = $this->transmitText($object, $contentStr);
}
return$resultStr;
```

代码执行后，效果展示如图4-9所示。

图4-9　同时使用消息接口和客服接口回复

4.4.3　客服接口的意义

客服接口是一个很有想像力的接口，它最大的特点在于可以和消息接口结合，同步和异步来使用不同的消息格式回复用户。而其他功能的加入以及相关技术的辅助，将有可能使得微信超越其他同类客服软件。

基于客服接口，我们能做出非常智能的客服系统。这些功能至少可以包括：能够实现电脑、手机双终端服务，随时随地提供客服服务；自动为不同问题或不同用户选择最专业的客服，达到最佳水平的服务；客服繁忙时可以将用户智能排队，或者设置用户的VIP等级来提供不同级别的服务响应速度；机器服务与人工服务同时进行，减少人工的工作量。

4.5　OAuth2.0网页授权

4.5.1　OAuth2.0

Auth是一个开放协议，允许用户让第三方应用以安全且标准的方式获取该用户在某一网站、移动或桌面应用上存储的私密的资源（如用户个人信息、照片、视频、联系人列表），而无需将用户名和密码提供给第三方应用。

OAuth 2.0是OAuth协议的升级版本，但不向后兼容OAuth 1.0。OAuth 2.0关注客户端开发者的简易性，同时为Web应用、桌面应用及手机和起居室设备提供专门的认证流程。

OAuth允许用户提供一个令牌，而不是用户名和密码来访问他们存放在特定服务提供者的数据。每一个令牌授权一个特定的网站（例如，视频编辑网站）在特定的时段（例如，接下来的2小时内）内访问特定的资源（例如仅仅是某一相册中的视频）。这样，OAuth允许用户授权第三方网站访问他们存储在另外的服务提供者上的信息，而不需要分享他们的访问许可或他们数据的所有内容。

OAuth官方网站：http://oauth.net/，OAuth2.0网址：http://oauth.net/2/。

4.5　OAuth2.0网页授权

4.5.1　OAuth2.0

Auth是一个开放协议，允许用户让第三方应用以安全且标准的方式获取该用户在某一网站、移动或桌面应用上存储的私密的资源（如用户个人信息、照片、视频、联系人列表），而无需将用户名和密

码提供给第三方应用。

OAuth 2.0是OAuth协议的升级版本，但不向后兼容OAuth 1.0。OAuth 2.0关注客户端开发者的简易性，同时为Web应用、桌面应用及手机和起居室设备提供专门的认证流程。

OAuth允许用户提供一个令牌，而不是用户名和密码来访问他们存放在特定服务提供者的数据。每一个令牌授权一个特定的网站（例如，视频编辑网站）在特定的时段（例如，接下来的2小时内）内访问特定的资源（例如仅仅是某一相册中的视频）。这样，OAuth允许用户授权第三方网站访问他们存储在另外的服务提供者上的信息，而不需要分享他们的访问许可或他们数据的所有内容。

OAuth官方网站：http://oauth.net/，OAuth2.0网址：http://oauth.net/2/。

4.5.2 授权过程

微信公众平台OAuth2.0授权详细步骤如下：

1）用户关注微信公众账号。

2）微信公众账号提供用户请求授权页面URL。

3）用户单击授权页面URL，向服务器发起请求。

4）服务器询问用户是否同意授权给微信公众账号（scope为snsapi_base时无此步骤）。

5）用户同意（scope为snsapi_base时无此步骤）。

6）服务器将code通过回调传给微信公众账号。

7）微信公众账号获得code。

8）微信公众账号通过code向服务器请求Access Token。

9）服务器返回Access Token和OpenID给微信公众账号。

10）微信公众账号通过Access Token向服务器请求用户信息（scope为snsapi_base时无此步骤）。

11）服务器将用户信息回送给微信公众账号（scope为snsapi_base时无此步骤）。

微信OAuth2.0授权基本流程如图4-10所示。

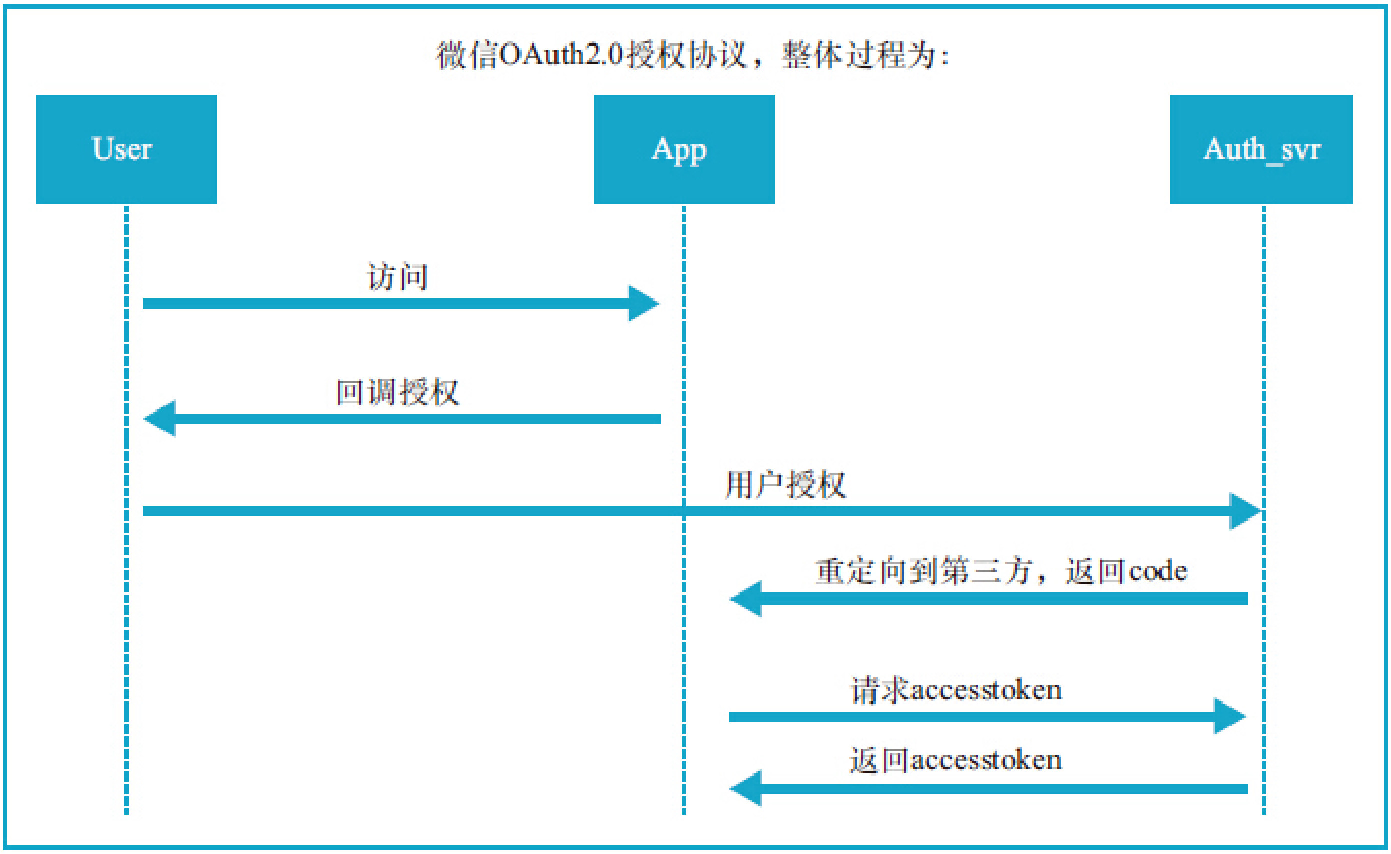


图4-10　微信OAuth2.0授权基本流程图

4.5.3 详细步骤

1.配置授权回调页面域名

在高级接口权限列表中，找到OAuth2.0网页授权，单击右侧的“修改”，如图4-11所示。

服务包	内容	操作	状态
基础接口	接收用户消息		有效
	向用户回复消息		
	接受事件推送		
自定义菜单	会话界面自定义菜单		有效
微信认证	普通认证		2014-11-07日到期
	搜索特权		
高级接口	语音识别(已开启)	关闭	2014-11-07日到期
	客服接口		
	OAuth2.0网页授权	修改	
	生成带参数二维码		
	获取用户地理位置(已关闭)	开启	
	获取用户基本信息		
	获取关注者列表		
	用户分组接口		
	上传下载多媒体文件		

图4-11　微信OAuth2.0授权

授权回调域名配置规范为全域名并且不带http，比如需要网页授权的域名为：www.qq.com，配置以后此域名下面的页面http://www.qq.com/music.html、http://www.qq.com/login.html都可以进行OAuth2.0鉴权。但http://pay.qq.com、http://music.qq.com、http://qq.com无法进行OAuth2.0鉴权。

下面我们填写mascot.duapp.com，它是方倍工作室的一个百度应用二级域名，如图4-12所示。

图4-12　配置授权回调域名

正常情况下，顶部应该提示“通过安全监测”，表示提交通过。

2.用户授权并获取code

在域名根目录下，新建一个文件，命名为oauth2.php，其内容为

```
<?php
if (isset($_GET['code'])){
echo$_GET['code'];
}else{
echo "NO CODE";
}
?>
```

先了解下请求授权页面的构造方式，接口请求如下：

```
https://open.weixin.qq.com/connect/oauth2/authorize?appid=APPID&redirect_uri=REDIRECT_URI&response_type=code&scope=SCOPE&state=STATE#wechat_redirect
```

请求授权参数说明如表4-12所示。

表4-12　授权请求参数说明

参数	是否必须	说明
appid	是	微信公众账号的唯一标识
redirect_uri	是	授权后重定向的回调链接地址
response_type	是	返回类型，请填写 code
scope	是	应用授权作用域，snsapi_base（不弹出授权页面，直接跳转，只能获取用户OpenID），snsapi_userinfo（弹出授权页面，可通过 OpenID 拿到昵称、性别、所在地。并且，即使在未关注的情况下，只要用户授权，也能获取其信息）
state	否	重定向后会带上 state 参数，开发者可以填写任意参数值
#wechat_redirect	否	直接在微信打开链接，可以不填此参数。做页面 302 重定向时，必须带此参数

构造请求接口如下：

· Scope为snsapi_userinfo（本节基于这个实现）

```
https://open.weixin.qq.com/connect/oauth2/authorize?appid=wx9a000b615d89c3f1&redirect_uri=http://mascot.duapp.com/oauth2.php&response_type=code&scope=snsapi_userinfo&state=1#we
```

· Scope为snsapi_base时

```
https://open.weixin.qq.com/connect/oauth2/authorize?appid=wx9a000b615d89c3f1&redirect_uri=http://mascot.duapp.com/oauth2.php&response_type=code&scope=snsapi_base&state=2#wechat
```

把这个链接发送或者回复到用户微信中，以便在微信浏览器中打开，这里使用A链接封装如下：

```
欢迎学习方倍工作室出品的微信公众平台开发教程
<a href="https://open.weixin.qq.com/connect/oauth2/authorize?appid=wx9a000b615d89c3f1&redirect_uri=http://mascot.duapp.com/oauth2.php&response_type=code&scope=snsapi_userinfo&s
```

在微信中显示如图4-13所示。

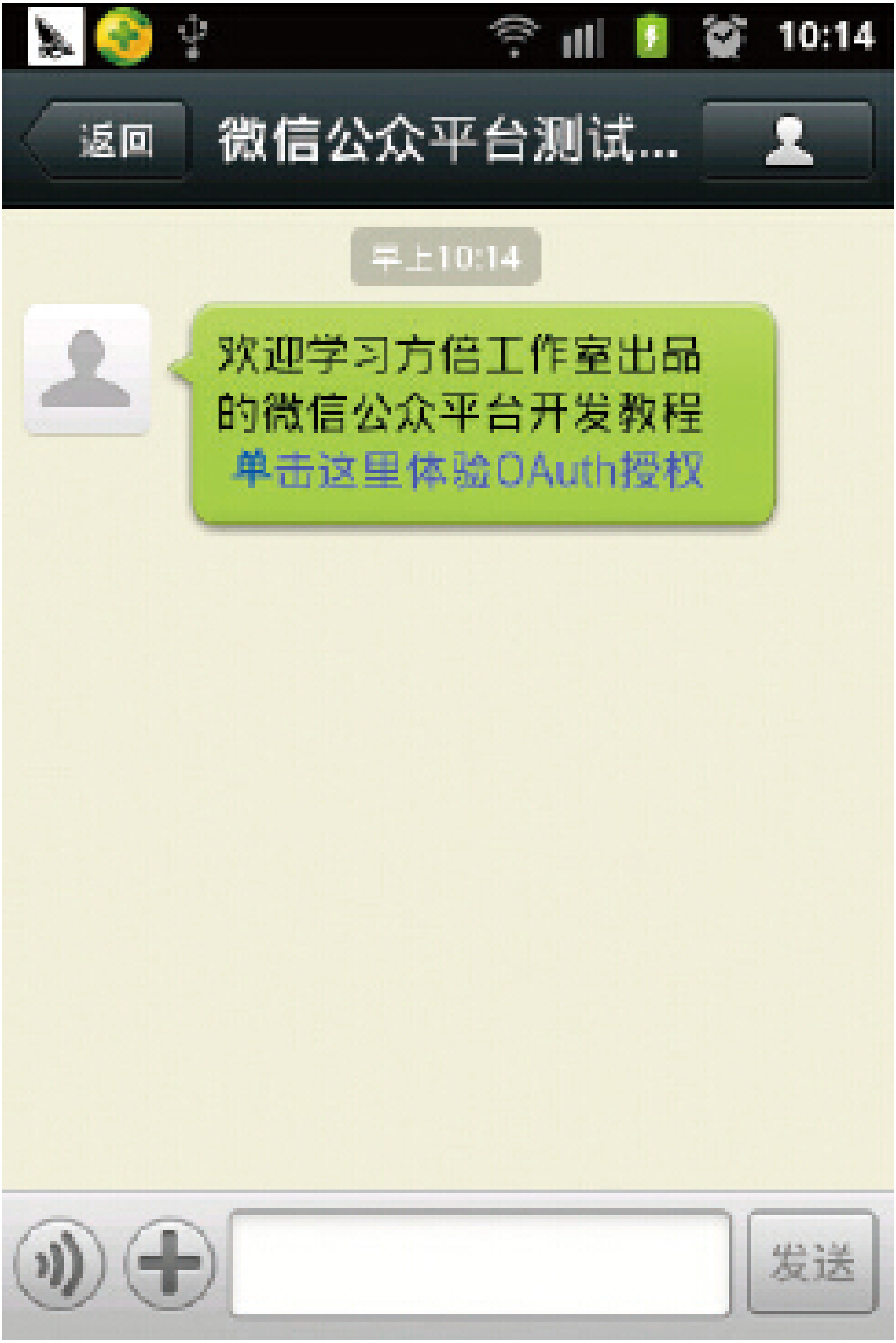

10:14
返回
微信公众平台测试...
早上10:14
欢迎学习方倍工作室出品的微信公众平台开发教程
单击这里体验OAuth授权
发送

图4-13　体验OAuth2授权

单击“单击这里体验OAuth授权”链接后，弹出应用授权界面，如图4-14所示。

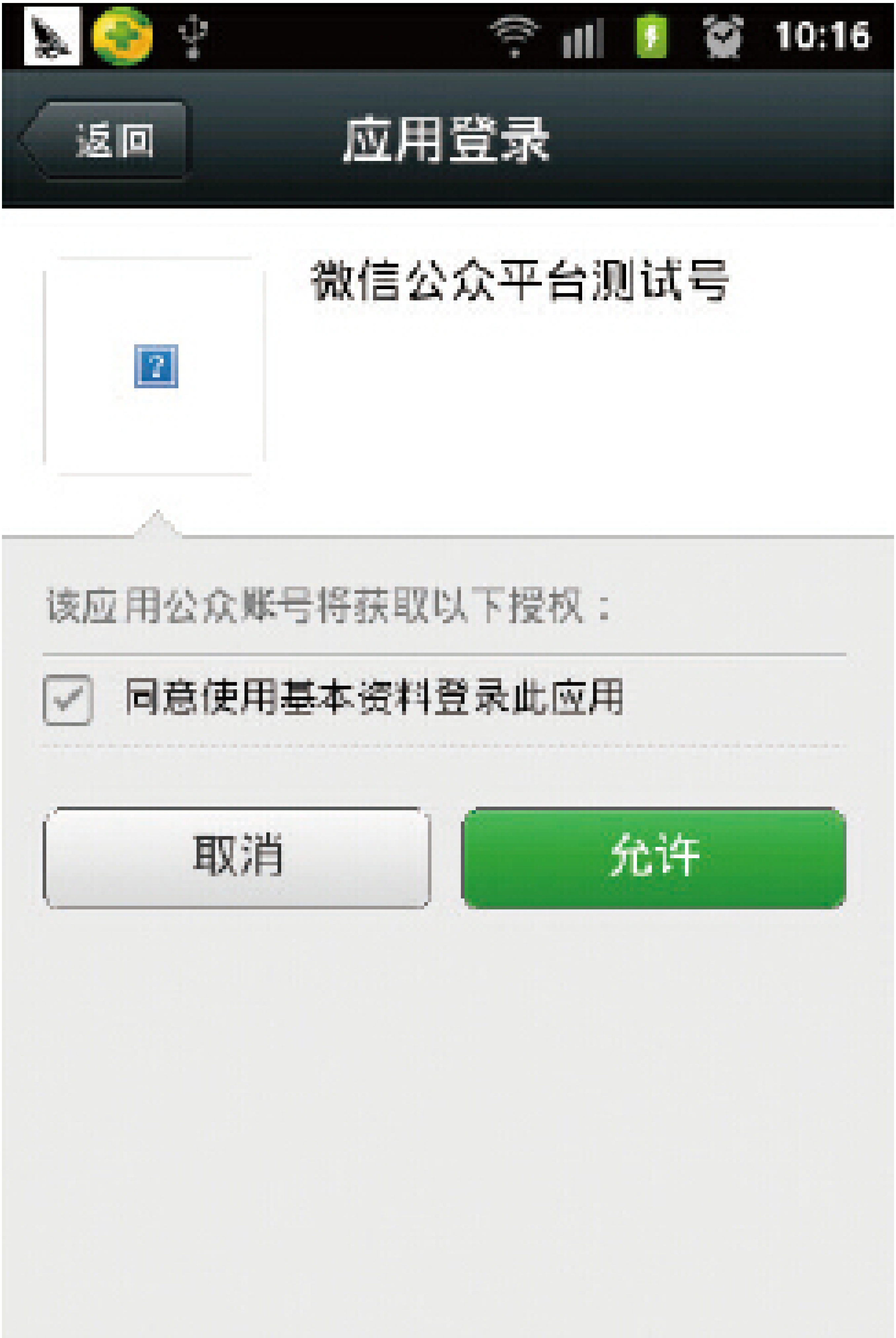

10:16
返回
应用登录
微信公众平台测试号
该应用公众账号将获取以下授权：
同意使用基本资料登录此应用
取消
允许

图4-14　应用登录

单击“允许”按扭，将跳转到oauth2.php页面，程序将获取到code，界面上显示如图4-15所示。

10:22
返回
应用登录
00364c54e24d0029f8db4274dcaff638

图4-15　获取code

这时候可以通过单击右上角按钮下的复制链接，得到如下链接：

```
http://mascot.duapp.com/oauth2.php?code=00364c54e24d0029f8db4274dcaff638&state=1
```

这样就成功得到了code。

3.使用code换取access_token

如果网页授权的作用域为snsapi_base，则本步骤中获取到网页授权access_token的同时，也获取到了openid，snsapi_base式的网页授权流程即到此为止。

换取网页授权access_token页面的接口如下：

```
https://api.weixin.qq.com/sns/oauth2/access_token?appid=APPID&secret=SECRET&code=CODE&grant_type=authorization_code
```

相关参数如表4-13所示。

表4-13　换取access_token请求参数说明

参数	是否必须	说明	参数	是否必须	说明
appid	是	微信公众账号的唯一标识	code	是	填写第一步获取的 code 参数
secret	是	微信公众账号的 appsecret	grant_type	是	填写为 authorization_code

构造请求如下：

```
https://api.weixin.qq.com/sns/oauth2/access_token?appid=wx9a000b615d89c3f1&secret=0ca77dd808ee1ea69830d7eecd770c06&code=00364c54e24d0029f8db4274dcaff638&grant_type=authorizatic
```

可以在浏览器中直接执行这条语句，得到如下JSON数据：

```
{
    "access_token": "OezXcEiiBSKSxW0eoylIeBxt9UjhFtdKikeq2gavEwzx1JhikLTyOhNThJV4l-qYxDZzhc7tEq4_4aNdgf12gPPO6-byBWFPPW0hS-_ElI7J3Pg7-gr4RqmBrY3fU1OOaJfd3tD7iU6qnGXX5f9UGA",
    "expires_in": 7200,
    "refresh_token": "OezXcEiiBSKSxW0eoylIeBxt9UjhFtdKikeq2gavEwzx1JhikLTyOhNThJV4l-qYxDZzhc7tEq4_4aNdgf12gNrlgJdOP1s3jqw49fpv-KQjni32A-DwXprScuG8_J2gJLqcT6WXH-fSFDr_Uk3NtA",
    "openid": "o7Lp5t6n59DeX3U0C7Kric9qEx-Q",
    "scope": "snsapi_userinfo,"
}
```

参数说明如表4-14所示。

表4-14　换取access_token结果参数说明

参数	说明
access_token	网页授权接口调用凭证，注意：此 access_token 与基础支持的 access_token 不同
expires_in	access_token 接口调用凭证超时时间，单位（秒）
refresh_token	用户刷新 access_token
openid	用户唯一标识，请注意，在未关注微信公众账号时，用户访问微信公众账号的网页，也会产生一个用户和微信公众账号唯一的 OpenID
scope	用户授权的作用域，使用逗号（,）分隔

于是，我们成功的通过code换取到了access_token，以及refresh_token。

•刷新access_token（如果需要）

官方文档中提到了刷新access_token的功能，但这不是必须要做的，初次学习可以先忽略。

由于access_token拥有较短的有效期，当access_token超时后，可以使用refresh_token进行刷新，refresh_token拥有较长的有效期（7天、30天、60天、90天），当refresh_token失效的后，需要用户重新授权。

获取上一步的refresh_token后，请求以下链接获取access_token：

请求接口如下：

```
https://api.weixin.qq.com/sns/oauth2/refresh_token?appid=APPID&grant_type=refresh_token&refresh_token=REFRESH_TOKEN
```

该接口参数说明如表4-15所示。

表4-15　刷新access_token请求参数说明

参数	是否必须	说明
appid	是	微信公众账号的唯一标识
grant_type	是	填写为 refresh_token
refresh_token	是	填写通过 access_token 获取到的 refresh_token 参数

这里构造如下URL：

```
https://api.weixin.qq.com/sns/oauth2/refresh_token?appid=wx9a000b615d89c3f1&grant_type=refresh_token&refresh_token=OezXcEiiBSKSxW0eoylIeBxt9UjhFtdKikeq2gavEwzx1JhikLTyOhNThJV4l
```

在浏览器中执行，可以得到前面同样格式的JSON数据。

4.使用access_token获取用户信息（scope为snsapi_userinfo时）

请求接口如下：

```
https://api.weixin.qq.com/sns/userinfo?access_token=ACCESS_TOKEN&openid=OPENID
```

该接口参数说明如表4-16所示。

表4-16　获取用户信息请求参数说明

参数	说明
access_token	网页授权接口调用凭证，注意：此 access_token 与基础支持的 access_token 不同
openid	用户的唯一标识
lang	返回国家地区语言版本，zh_CN 简体，zh_TW 繁体，en 英语

构造请求如下：

```
https://api.weixin.qq.com/sns/userinfo?access_token=OezXcEiiBSKSxW0eoylIeBxt9UjhFtdKikeq2gavEwzx1JhikLTyOhNThJV4l-qYxDZzhc7tEq4_4aNdgf12gPPO6-byBWFPPW0hS-_ElI7J3Pg7-gr4RqmBrY3f
```

执行上述接口后，得到如下数据：


```
{
"openid": "o7Lp5t6n59DeX3U0C7Kric9qEx-Q",
"nickname": "方倍",
"sex": 1,
"language": "zh_CN",
"city": "Shenzhen",
"province": "Guangdong",
"country": "CN",
"headimgurl": "http://wx.qlogo.cn/mmopen/Kkv3HV30gbEZmoo1rTrP4UjRRqzsibUjT9JClPJy3gzo0NkEqzQ9yTSJzErnsRqoLIct5NdLJgcDMicTEBiaibzLn34JLwficVvl6/0",
"privilege": []
}
```


上述参数说明如表4-17所示。

表4-17　用户信息结果参数说明

参数	说明
openid	用户的唯一标识
nickname	用户昵称
sex	用户的性别，值为 1 时是男性，值为 2 时是女性，值为 0 时是未知
province	用户个人资料填写的省份
city	普通用户个人资料填写的城市
country	国家，如中国为 CN
headimgurl	用户头像，最后一个数值代表正方形头像大小（有 0、46、64、96、132 数值可选，0 代表 640*640 正方形头像），用户没有头像时该项为空
privilege	用户特权信息，json 数组，如微信沃卡用户为（chinaunicom）

这与作者的个人微信是一致的，如图4-16所示。

11:14
返回
个人信息
头像
名字
方倍
二维码名片
我的收货地址
性别
男
地区
广东 深圳
个性签名
发上等愿结中等缘享下等福
择高处立寻平处住向宽处行

图4-16　个人信息

至此，在不输入账号及密码的情况下，微信公众账号获得了用户的个人信息，这些信息包括昵称、性别、国家、省份、城市、个人头像以及特权列表。

一个完整的OAuth2认证就完成了。

4.5.4　效果展示

以下PHP代码实现了获取用户信息的功能。

注意，域名配置以及回调URL均需要配置正确才能成功获得用户信息。

```
$code = $_GET["code"];
$userinfo = getUserInfo($code);
function getUserInfo($code)
{
    $appid = "wx9a000b615d89c3f1";
    $appsecret = "0ca77dd808ee1ea69830d7eecd770c06";
    $access_token = "";
    //根据code获得Access Token
    $access_token_url = "https://api.weixin.qq.com/sns/oauth2/access_token?appid=$appid&secret=$appsecret&code=$code&grant_type=authorization_code";
    $access_token_json = https_request($access_token_url);
    $access_token_array = json_decode($access_token_json, true);
    $access_token = $access_token_array['access_token'];
    $openid = $access_token_array['openid'];
    //根据Access Token和OpenID获得用户信息
    $userinfo_url = "https://api.weixin.qq.com/sns/userinfo?access_token=$access_token&openid=$openid ";
    $userinfo_json = https_request($userinfo_url);
    $userinfo_array = json_decode($userinfo_json, true);
    return $userinfo_array;
}
function https_request($url)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $data = curl_exec($curl);
    if (curl_errno($curl)) {return 'ERROR '.curl_error($curl);}
    curl_close($curl);
    return $data;
}
```

在jQuery Mobile中列出用户信息，HTML核心代码如下所示：

```
<div class="fieldcontain">
    <label for="openid">OpenID</label>
    <input name="openid" id="openid" value="<?php echo $userinfo ["openid"]; ?>" type="text" >
</div>
<div class="fieldcontain">
    <label for="nickname">昵称</label>
    <input name="nickname" id="nickname" value="<?php echo $userinfo ["nickname"];?>" type="text" >
</div>
<div class="fieldcontain">
    <label for="sex">性别</label>
    <input name="sex" id="sex" value="<?php echo $userinfo["sex"];?>" type="text" >
</div>
<div class="fieldcontain">
    <label for="country">国家</label>
    <input name="country" id="country" value="<?php echo $userinfo ["country"];?>" type="text" >
</div>
<div class="fieldcontain">
    <label for="province">省份</label>
    <input name="province" id="province" value="<?php echo $userinfo ["province"];?>" type="text" >
</div>
<div class="fieldcontain">
    <label for="city">城市</label>
    <input name="city" id="city" value="<?php echo $userinfo["city"];?>" type="text" >
</div>
<div class="fieldcontain">
    <label for="privilege">特权</label>
    <input name="privilege" id="privilege" value="<?php echo $userinfo ["privilege"];?>" type="text" >
</div>
```

效果展示如图4-17所示。

图4-17　OAuth2.0获取个人信息

4.6　生成带参数二维码

为了满足用户渠道推广分析的需要，公众平台提供了生成带参数二维码的接口。使用该接口可以获得多个带不同场景值的二维码，用户扫描后，微信公众账号可以接收到事件推送。

4.6.1　场景二维码

目前有两种类型的二维码，分别是临时二维码和永久二维码，前者有过期时间，最长时间为1800秒，但能够生成较多数量，后者无过期时间，数量较少（目前参数只支持1~100000）。两种二维码分别适用于账号绑定、用户来源统计等场景。

4.6.2　创建二维码ticket

每次创建二维码ticket需要提供一个开发者自行设定的参数（scene_id），下面将分别介绍临时二维码和永久二维码的创建二维码ticket过程。

1.临时二维码

请求接口如下：

```
https://api.weixin.qq.com/cgi-bin/qrcode/create?access_token=TOKEN
```

该接口的POST数据格式如下，参数说明如表4-18所示。

表4-18　生成二维码请求参数说明

参数	说明
expire_seconds	该二维码有效时间，以秒为单位，最长时间不超过 1800 秒（30 分钟）
action_name	二维码类型，QR_SCENE 为临时 ,QR_LIMIT_SCENE 为永久
action_info	二维码详细信息
scene_id	场景值 ID，临时二维码时为 32 位非 0 整型，永久二维码时最大值为 100 000（目前参数只支持 1 ～ 100 000）

```
{
    "expire_seconds": 1800,
    "action_name": "QR_SCENE",
    "action_info": {
            "scene": {
                    "scene_id": 100000
        }
    }
}
```

返回格式如下，参数说明如表4-19所示。

表4-19　生成二维码结果参数说明

参数	说明
ticket	获取的二维码 ticket，凭借此 ticket 可以在有效时间内换取二维码
expire_seconds	二维码的有效时间，以秒为单位，最长时间不超过 1800（永久二维码中无此参数）

```
{
    "ticket": "gQFK8DoAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL3kweXE0T3JscWY3UTltc3ZPMklvAAIEG9jUUgMECAcAAA==",
    "expire_seconds": 1800
}
```

2.永久二维码

请求接口如下：

```
https://api.weixin.qq.com/cgi-bin/qrcode/create?access_token=TOKEN
```

该接口的POST数据格式如下，参数说明如表4-18所示。

```
{
    "action_name": "QR_LIMIT_SCENE",
    "action_info": {
            "scene": {
                    "scene_id": 1000
        }
    }
}
```

返回格式如下，参数说明如表4-19所示。

```
{
    "ticket": "gQHi8DoAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL0UweTNxNi1sdlA3RklyRnNKbUFvAAIELdnUUgMEAAAAAA=="
}
```

创建二维码的程序实现如下：

```
$access_token = "xDx0pD_ZvXkHM3oeu5oGjDt1_9HxlA-9g0vtR6MZ-v4r7MpvZYC4ee4OxN97Lr4irkPKE94tzBUhpZG_OvqAC3D3XaWJIGIn0eeIZnfaofO1C3LNzGphd_rEv3pIimsW9lO-4FOw6D44T3sNsQ5yXQ";
//临时
$qrcode = '{"expire_seconds": 1800, "action_name": "QR_SCENE", "action_info": {"scene": {"scene_id": 10000}}}';
//永久
$qrcode = '{"action_name": "QR_LIMIT_SCENE", "action_info": {"scene": {"scene_id": 1000}}}';
$url = "https://api.weixin.qq.com/cgi-bin/qrcode/create?access_token=$access_token";
$result = https_request($url,$qrcode);
$jsoninfo = json_decode($result, true);
$ticket = $jsoninfo["ticket"];
function https_request($url, $data = null){
$curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
$output = curl_exec($curl);
    curl_close($curl);
return$output;
}
```

4.6.3　通过ticket换取二维码

获取二维码ticket后，开发者可用ticket换取二维码图片。该操作无须登录或授权即可调用。

换取二维码请求接口如下（注意TICKET必需UrlEncode）：

```
https://mp.weixin.qq.com/cgi-bin/showqrcode?ticket=TICKET
```

ticket正确情况下，http返回码是200，是一张图片，可以直接展示或者下载。

上述接口运行后，返回的HTTP头示例如下所示，其中的url即二维码的图片地址。

```
{
    "url": "https://mp.weixin.qq.com/cgi-bin/showqrcode?ticket=gQHi8DoAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL0UweTNxNi1sdlA3RklyRnNKbUFvAAIELdnUUgMEAAAAAA%3D%3D",
    "content_type": "image/jpg",
    "http_code": 200,
    "header_size": 162,
    "request_size": 181,
    "filetime": -1,
    "ssl_verify_result": 20,
    "redirect_count": 0,
    "total_time": 0.509,
    "namelookup_time": 0,
    "connect_time": 0.058,
    "pretransfer_time": 0.343,
    "size_upload": 0,
    "size_download": 28497,
    "speed_download": 55986,
    "speed_upload": 0,
    "download_content_length": 28497,
    "upload_content_length": 0,
    "starttransfer_time": 0.481,
    "redirect_time": 0
}
```

下面分别是两种场景二维码的URL：

```
https://mp.weixin.qq.com/cgi-bin/showqrcode?ticket=gQFK8DoAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL3kweXE0T3JscWY3UTltc3ZPMklvAAIEG9jUUgMECAcAAA%3d%3d
https://mp.weixin.qq.com/cgi-bin/showqrcode?ticket=gQHi8DoAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL0UweTNxNi1sdlA3RklyRnNKbUFvAAIELdnUUgMEAAAAAA%3d%3d
```

上述URL代码的两种二维码图片如图4-18所示。

图4-18　临时二维码和永久二维码

4.6.4　下载二维码

二维码生成以后，可以在浏览器中右击另存为本地图片，但如果有很多二维码的时候，用程序来下载就比较方便。

我们使用CURL获取图片的所有信息，并将图片数据保存为一个文件。一个完整的下载代码如下：

```
$ticket = "gQHi8DoAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL0UweTNxNi1sdlA3RklyRnNKbUFvAAIELdnUUgMEAAAAAA==";
$url = "https://mp.weixin.qq.com/cgi-bin/showqrcode?ticket=".urlencode($ticket);
$imageInfo = downloadWeixinFile($url);
$filename = "qrcode.jpg";
$local_file = fopen($filename, 'w');
if (false !== $local_file){
    if (false !== fwrite($local_file, $imageInfo["body"])) {
        fclose($local_file);
    }
}
function downloadWeixinFile($url)
{
```

```
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_NOBODY, 0);    //只取body头
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $package = curl_exec($ch);
    $httpinfo = curl_getinfo($ch);
    curl_close($ch);
    return array_merge(array('body' => $package), array('header' => $httpinfo));
}
```

这样，在程序当前目录就会生成一个包含二维码的图片文件。

4.6.5 扫描带参数二维码事件

用户扫描带场景值二维码时，可能推送以下两种事件：

· 如果用户还未关注微信公众账号，则用户可以关注微信公众账号，关注后微信会将带场景值关注事件推送给开发者。

· 如果用户已经关注微信公众账号，则微信会将带场景值扫描事件推送给开发者。

1.用户未关注时

进行关注后的事件推送的XML数据格式如下：

```
<xml>
    <ToUserName><![CDATA[gh_45072270791c]]></ToUserName>
    <FromUserName><![CDATA[o7Lp5t6n59DeX3U0C7Kric9qEx-Q]]></FromUserName>
    <CreateTime>1389684286</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[subscribe]]></Event>
    <EventKey><![CDATA[qrscene_1000]]></EventKey>
    <Ticket><![CDATA[gQHi8DoAAAAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL0UweTNxNi1sdlA3RklyRnNKbUFvAAIELdnUUgMEAAAAAA==]]></Ticket>
</xml>
```

上述数据的参数说明如表4-20所示。

表4-20　未关注时事件推送参数说明

参数	说明
ToUserName	微信公众账号微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	消息类型，event
Event	事件类型，subscribe
EventKey	事件 KEY 值，qrscene_ 为前缀，后面为二维码的参数值
Ticket	二维码的 ticket，可用来换取二维码图片

2.用户已关注时

事件推送的XML数据如下所示：

```
<xml>
    <ToUserName><![CDATA[gh_45072270791c]]></ToUserName>
   <FromUserName><![CDATA[o7Lp5t6n59DeX3U0C7Kric9qEx-Q]]></FromUserName>
   <CreateTime>1389684184</CreateTime>
   <MsgType><![CDATA[event]]></MsgType>
   <Event><![CDATA[SCAN]]></Event>
   <EventKey><![CDATA[1000]]></EventKey>
   <Ticket><![CDATA[gQHi8DoAAAAAAAAAAAAASxodHRwOi8vd2VpeGluLnFxLmNvbS9xL0UweTNxNi1sdlA3RklyRnNKbUFvAAIELdnUUgMEAAAAAA==]]></Ticket>
</xml>
```

上述数据的参数说明如表4-21所示。

表4-21　已关注时事件推送参数说明

参数	说明
ToUserName	微信公众账号微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	消息类型，event
Event	事件类型，scan
EventKey	事件 KEY 值，是一个 32 位无符号整数，即创建二维码时的二维码 scene_id
Ticket	二维码的 ticket，可用来换取二维码图片

以下代码判定了扫描带参数二维码的两种情形：

```
private function receiveEvent($object)
{
    $contentStr = "";
    switch ($object->Event)
    {
        case "subscribe":
            $contentStr = "欢迎关注 ";
            if (isset($object->EventKey)){
                $contentStr .= "\n来自二维码场景 ".$object->EventKey;
            }
            break;
        case "SCAN":
            $contentStr = "扫描二维码场景 ".$object->EventKey;
            break;
        default:
            break;
    }
    $resultStr = $this->transmitText($object, $contentStr);
    return $resultStr;
}
```

4.7 获取用户地理位置

开通了上报地理位置接口的微信公众账号，在用户关注并进入微信公众账号会话时，会弹框让用户确认是否允许微信公众账号使用其地理位置。弹框只在关注后出现一次，用户以后可以在微信公众账号详情页面进行操作。

4.7.1 获取用户地理位置的方式

获取用户地理位置的方式有两种，一种是仅在进入会话时上报一次，一种是进入会话后每隔5秒上报一次。微信公众账号可以在公众平台网站中设置。

用户同意上报地理位置后，每次进入微信公众账号会话时，都会在进入时上报地理位置，或在进入会话后每5秒上报一次地理位置，上报地理位置以推送XML数据包到开发者填写的URL来实现。

推送XML数据包示例如下：

```
<xml>
    <ToUserName><![CDATA[gh_45072270791c]]></ToUserName>
    <FromUserName><![CDATA[o7Lp5t5BZDl22PcjIliHp03kzgBE]]></FromUserName>
    <CreateTime>1389686084</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[LOCATION]]></Event>
    <Latitude>28.551088</Latitude>
    <Longitude>112.123856</Longitude>
    <Precision>96.000000</Precision>
</xml>
```

上述数据的参数说明如表4-22所示。

表4-22　获取用户位置信息参数说明

参数	说明
ToUserName	开发者微信号
FromUserName	发送方账号（一个 OpenID）
CreateTime	消息创建时间（整型）
MsgType	消息类型，event
Event	事件类型，LOCATION
Latitude	地理位置纬度
Longitude	地理位置经度
Precision	地理位置精度

4.7.2 转换坐标到地址

我们先了解一下坐标体系的相关知识，目前国内外主要有以下几种不同的坐标体系。

· WGS-84坐标系（World Geodetic System-1984 Coordinate System）是一种国际上采用的地心坐标系。坐标原点为地球质心，其地心空间直角坐标系的Z轴指向BIH（国际时间服务机构）1984.O定义的协议地球极（CTP）方向，X轴指向BIH 1984.0的零子午面和CTP赤道的交点，Y轴与Z轴、X轴垂直构成右手坐标系，称为1984年世界大地坐标系统。该坐标系统俗称为“地球坐标系”，即目前GPS所采用的坐标系统。

· GCJ-02是由中国国家测绘局制订的地理信息系统的坐标系统。它是一种对经纬度数据的加密算法，即加入随机的偏差。国内出版的各种地图系统（包括电子形式），必须至少采用GCJ-02对地理位置进行首次加密。该坐标系统也俗称为“火星坐标系”。

· 百度坐标系统是百度公司的地图坐标系统，它在国家测绘局的GCJ-02的基础上，进行了BD-09二次加密，更加保护个人隐私。同时，百度地图支持将WGS-84坐标和GCJ-02坐标转换成百度坐标。

当微信公众账号获取到用户地理位置坐标以后，需要将坐标转换成用户的真实所在地（省、市、区等信息），以便于更直观地理解以及进行用户数据分析。百度地图Geocoding API提供逆地址解析功能，即由经纬度信息得到地址信息，使用这一功能可以获得微信用户的具体所在地。该接口支持上述三种坐标系统的坐标，在不同的坐标系统中，经纬度解析后的地址有一定的偏差。例如：“lat:

22.539968，lng：113.954980”在百度坐标下解析的结果是“广东省深圳市南山区高新南七道2号”，而在GCJ-02坐标系下解析的结果为“广东省深圳市南山区深南大道9789-2”。

在微信5.2版本中，安卓版上使用的地图为腾讯地图，苹果版上使用的地图为Google中国地图，两种版本均采用GCJ-02坐标系统。当我们进行逆地址解析的时候，需要注意选择正确的坐标类型。

百度地图Geocoding API服务地址接口为：

```
http://api.map.baidu.com/geocoder/v2/
```

该接口的参数说明如表4-23所示。

表4-23　百度地图Geocoding API参数说明

参数	是否必须	默认值	格式举例	含义
output	否	xml	json	输出格式为 JSON 或者 XML
ak	是	无	E4805d16520de693a3fe707cdc962045	用户申请注册的 key，自 v2 开始参数修改为“ak”，之前版本参数为“key”
callback	否	无	callback=showLocation (JavaScript 函数名)	将 json 格式的返回值通过 callback 函数返回以实现 jsonp 功能
coordtype	否	bd09ll	gcj02ll 国测局经纬度坐标	坐标的类型，目前支持的坐标类型包括：bd09ll（百度经纬度坐标）、gcj02ll（国测局经纬度坐标）、wgs84ll（GPS 经纬度），微信上的地图采用的为 gcj02ll
location	是	无	38.76623,116.43213 lat< 纬度 >,lng< 经度 >	根据经纬度坐标获取地址
pois	否	0	0	是否显示指定位置周边的 poi，0 为不显示，1 为显示。当值为 1 时，显示周边 100 米内的 poi

该接口的一个请求示例如下：

```
http://api.map.baidu.com/geocoder/v2/?ak=B944e1fce373e33ea4627f95f54f2ef9&location
```

上述接口执行后，返回结果如下：

```
{
    "status": 0,
    "result": {
        "location": {
            "lng": 113.96151089462,
            "lat": 22.54572267759
        },
        "formatted_address": "广东省深圳市南山区深南大道9789-2",
        "business": "科技园,大冲,深圳湾",
        "addressComponent": {
            "city": "深圳市",
            "district": "南山区",
            "province": "广东省",
            "street": "深南大道",
            "street_number": "9789-2"
        },
        "cityCode": 340
    }
}
```

在微信中接收到用户地理位置事件的时候，可以把用户的坐标转换成国内的省市区地址，核心代码如下：

```
private function receiveEvent($object)
{
    $contentStr = "";
    switch ($object->Event)
    {
        case "subscribe":
        $contentStr = "欢迎关注".(isset($object->EventKey)?("\n场景 ".$object->EventKey):"");
            break;
        case "unsubscribe":
            $contentStr = "取消关注";
            break;
        case "LOCATION":
            $url = "http://api.map.baidu.com/geocoder/v2/?ak=B944e1fce373e33ea4627f95f54f2ef9&location=$object->Latitude,$object->Longitude&output=json&coordtype=gcj02ll";
            $output = file_get_contents($url);
            $address = json_decode($output, true);
            $contentStr = "位置 ".$address["result"]["addressComponent"]["province"]."
 ".$address["result"]["addressComponent"]["city"]." ".$address["result"]["addressComponent"]["district"]." ".$address["result"]["addressComponent"]["street"];
          break;
        default:
            break;
    }
    $resultStr = $this->transmitText($object, $contentStr);
    return $resultStr;
}
```

这样，我们就能实时收集到很多用户的实际位置。把这些信息保存到数据库，我们还可以对用户做大数据分析。

4.7.3　用户地理位置的意义

获得用户的地理位置能帮助企业更加了解它们的用户。根据位置数据能够了解用户在什么时候、什么地方、有什么样的生活习惯，并且追踪群发事件的效果。

位置数据分析功能能够预测用户的行为，从而为公司的经营策略制定提供依据。这是件很有意义的事情，但是，这其实才展示了移动位置数据分析的冰山一角。现在我们知道用户在何处，但在未来我们也许能够知道用户在做什么、在想什么，以及更多。

4.8　获取用户基本信息

在关注者与微信公众账号产生消息交互后，微信公众账号可获得关注者的OpenID（加密后的微信号，每个用户对每个微信公众账号的OpenID是唯一的。对于不同微信公众账号，同一用户的OpenID不同）。微信公众账号可通过本接口来根据OpenID获取用户基本信息，包括昵称、头像、性别、所在城市、语言和关注时间。

4.8.1　获取用户基本信息的方式

我们可通过OpenID来获取用户基本信息。

获取用户信息的接口如下：

```
https://api.weixin.qq.com/cgi-bin/user/info?access_token=ACCESS_TOKEN&openid=OPENID&lang=zh_CN
```

该接口的参数说明如表4-24所示。

表4-24　获取用户信息请求参数说明

参数	是否必须	说明
access_token	是	调用接口凭证
openid	是	普通用户的标识，对当前微信公众账号唯一
lang	否	返回国家地区语言版本，zh_CN 简体，zh_TW 繁体，en 英语

可以构造URL如下：

```
https://api.weixin.qq.com/cgi-bin/user/info?access_token=5ic8RoqZd7IBUtE0aGXUhcdKBnrwlO1iqtvyJJUgIcDcopJIH7x90QH4yiK_z08fqp4WOD7kfKPDtET29BRYNZO-sSIITaxIj4u72LYAYcFH6t1fiSVzjUu
```

获取用户基本信息的代码实现如下所示：

```
$access_token = "5ic8RoqZd7IBUtE0aGXUhcdKBnrwlO1iqtvyJJUgIcDcopJIH7x90QH4yiK_z08fqp4WOD7kfKPDtET29BRYNZO-sSIITaxIj4u72LYAYcFH6t1fiSVzjUuqYWOtHiw0LGDorxx5Qj6SN0Z7GtYGTA";
$openid="o7Lp5t6n59DeX3U0C7Kric9qEx-Q";
$url = "https://api.weixin.qq.com/cgi-bin/user/info?access_token=$access_token&openid=$openid&lang=zh_CN";
$output = https_request($url);
var_dump($output);
function https_request($url)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $data = curl_exec($curl);
    if (curl_errno($curl)) {return 'ERROR '.curl_error($curl);}
    curl_close($curl);
    return $data;
}
```

上述代码执行后，返回结果如下：

```
{
    "subscribe": 1,
    "openid": "o7Lp5t6n59DeX3U0C7Kric9qEx-Q",
    "nickname": "方倍",
    "sex": 1,
    "language": "zh_CN",
    "city": "深圳",
    "province": "广东",
    "country": "中国",
    "headimgurl": "http://wx.qlogo.cn/mmopen/Kkv3HV30gbEZmoo1rTrP4UjRRqzsibUjT9JClPJy3gzo0NkEqzQ9yTSJzErnsRqoLIct5NdLJgcDMicTEBiaibzLn34JLwficVvl6/0",
    "subscribe_time": 1389684286
}
```

上述数据的参数说明如表4-25所示。

表4-25　获取用户信息结果参数说明

参数	说明
subscribe	用户是否订阅该微信公众账号标识，值为 0 时，代表此用户没有关注该微信公众账号，拉取不到其余信息
openid	用户的标识，对当前微信公众账号唯一
nickname	用户的昵称
sex	用户的性别，值为 1 时是男性，值为 2 时是女性，值为 0 时是未知
city	用户所在城市
country	用户所在国家
province	用户所在省份
language	用户的语言，简体中文为 zh_CN
headimgurl	用户头像，最后一个数值代表正方形头像大小（有 0、46、64、96、132 数值可选，0 代表 640 × 640 正方形头像），用户没有头像时该项为空
subscribe_time	用户关注时间，为时间戳。如果用户曾多次关注，则取最后关注时间

4.8.2 制作个性欢迎语

我们可以根据用户的基本信息来制作更有个性的欢迎语。

以下代码实现了关注时取用户基本信息的功能，能在用户关注时返回用户的基础信息，当然我们可以做些修改调整，使欢迎语更接地气。

```
private function receiveEvent($object)
{
    $contentStr = "";
    switch ($object->Event)
    {
        case "subscribe":
        $access_token = "nFX6GFsspSLBKJLgMQ3kj1YM8_FchRE7vE2ZOIlmfiAB0WZh8c5xcCRiXGLL3h3uUamBOb7Tr7cix_A-UQwig9_-mohVK338yJG6wnv_cvVCZj6PQBjTuT_yKbiuIceKYqFhCjMFU21Z_b7d6Go88w"
            $openid = $object->FromUserName;
            $url = "https://api.weixin.qq.com/cgi-bin/user/info?access_token=$access_
token&openid=$openid&lang=zh_CN";
            $output = $this->https_request($url);
            $jsoninfo = json_decode($output, true);
            $contentStr = "您好，".$jsoninfo["nickname"]."\n".
            "性别：".(($jsoninfo["sex"] == 1)?"男":(($jsoninfo["sex"] == 2)?"女":"未知"))."\n".
          "地区：".$jsoninfo["country"]." ".$jsoninfo["province"]." ".$jsoninfo["city"]."\n".
           "语言：".(($jsoninfo["language"] == "zh_CN")?"简体中文":"非简体中文")."\n".
            "关注：".date('Y年m月d日',$jsoninfo["sub
scribe_time"]);
            break;
    }
    $resultStr = $this->transmitText($object, $contentStr);
    return $resultStr;
}
```

上述代码运行时，效果展示如图4-19所示。

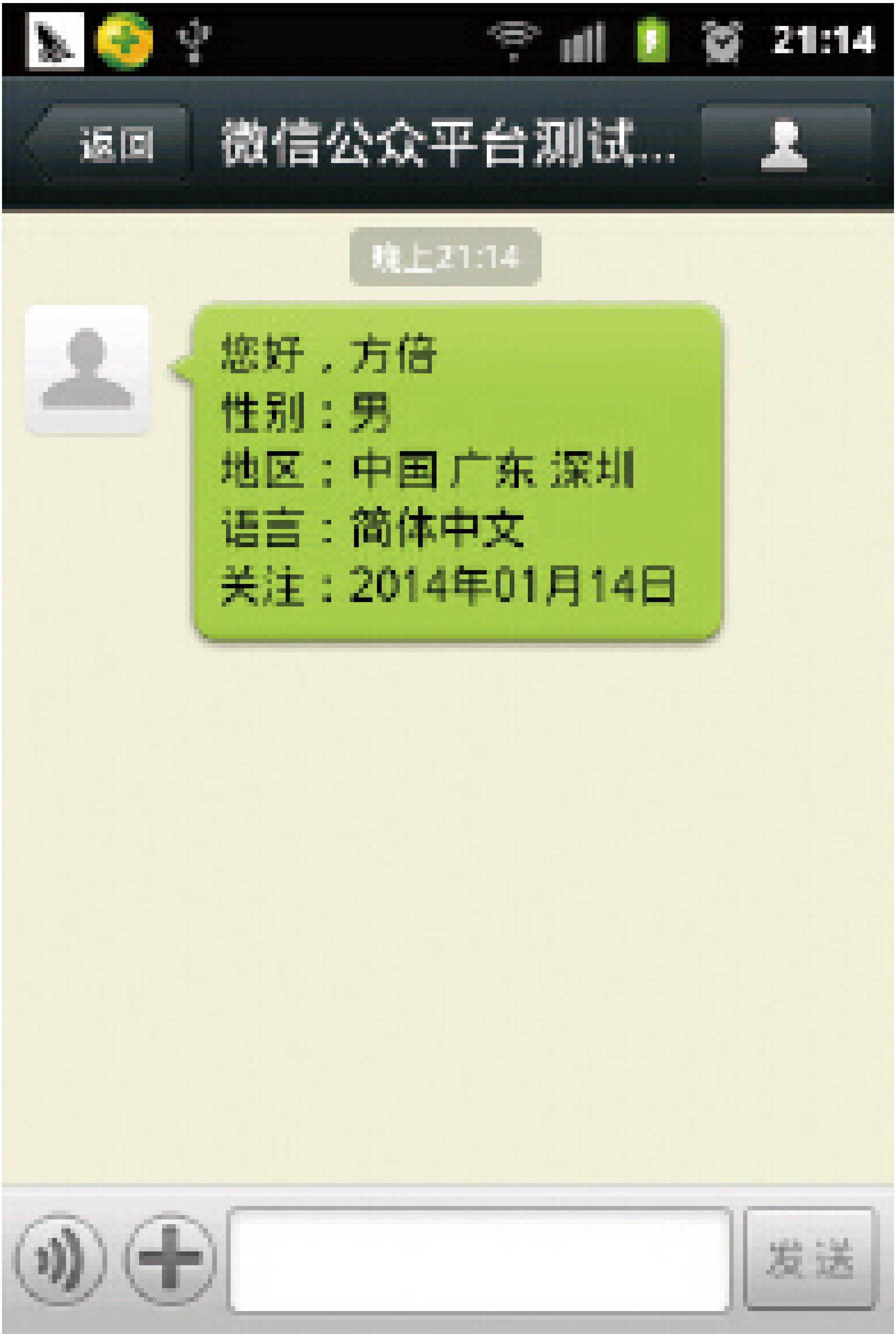


图4-19　用户基本信息欢迎

4.9 获取关注者列表

微信公众账号可通过本接口来获取账号的关注者列表，关注者列表由一串OpenID（加密后的微信号，每个用户对每个微信公众账号的OpenID是唯一的）组成。一次拉取调用最多拉取10000个关注者的OpenID，可以通过多次拉取的方式来满足需求。

4.9.1 获取关注者列表的方式

接口调用请求说明如下：

```
https://api.weixin.qq.com/cgi-bin/user/get?access_token=ACCESS_TOKEN&next_openid=NEXT_OPENID
```

上述接口参数如表4-26所示。

表4-26 获取关注者列表请求参数说明

参数	是否必须	说明
access_token	是	调用接口凭证
next_openid	是	第一个拉取的 OPENID，不填默认从头开始拉取

构造请求列表如下：

```
https://api.weixin.qq.com/cgi-bin/user/get?access_token=5ic8RoqZd7IBUtE0aGXUhcdKBnrwlO1iqtvyJJUgIcDcopJIH7x90QH4yiK_z08fqp4WOD7kfKPDtET29BRYNZO-sSIITaxIj4u72LYAYcFH6t1fiSVzjUuc
```

获取关注者列表的代码实现如下所示：

```
$access_token = "";
$url = "https://api.weixin.qq.com/cgi-bin/user/get?access_token= $access_token";
$result = https_request($url);
$jsoninfo = json_decode($result, true);
var_dump($result);
function https_request($url)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $data = curl_exec($curl);
    if (curl_errno($curl)) {return 'ERROR '.curl_error($curl);}
    curl_close($curl);
    return $data;
}
```

代码执行后，返回结果如下所示：

```
{
    "total": 200,
    "count": 200,
    "data": {
        "openid": [
            "o7Lp5t6n59DeX3U0C7Kric9qEx-Q",
            "o7Lp5t5BZDl22PcjIliHp03kzgBE"
        ]
    },
    "next_openid": "o7Lp5t5BZDl22PcjIliHp03kzgBE"
}
```

返回结果参数说明如表4-27所示。

表4-27 获取关注者列表结果参数说明

参数	说明
total	关注该微信公众账号的总用户数
count	拉取的 OpenID 个数，最大值为 10 000
data	列表数据，OpenID 的列表
next_openid	拉取列表的后一个用户的 OpenID

4.9.2 分批获取

当微信公众账号关注者数量超过10000时，可通过填写next_openid的值，从而多次拉取列表的方式来满足需求。

具体而言，就是在调用接口时，将上一次调用得到的返回中的next_openid值作为下一次调用中的next_openid值。

示例如下。

微信公众账号A拥有23000个关注的人，想通过拉取关注接口获取所有关注的人，那么需要三次请求，下面将分别说明。

第一次请求的URL：

```
https://api.weixin.qq.com/cgi-bin/user/get?access_token=ACCESS_TOKEN
```

返回结果如下：

```
{
  "total":23000,
  "count":10000,
  "data":{"
     openid":[
        "OPENID1",
        "OPENID2",
        http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/...,
        "OPENID10000"
     ]
   },
   "next_openid":"NEXT_OPENID1"
}
```

第二次请求的URL：

```
https://api.weixin.qq.com/cgi-bin/user/get?access_token=ACCESS_TOKEN&next_openid=NEXT_OPENID1
```

返回结果如下：

```
{
   "total":23000,
   "count":10000,
   "data":{
     "openid":[
       "OPENID10001",
       "OPENID10002",
       http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/...,
       "OPENID20000"
     ]
   },
   "next_openid":"NEXT_OPENID2"
}
```

第三次请求URL：

```
https://api.weixin.qq.com/cgi-bin/user/get?access_token=ACCESS_TOKEN&next_openid=NEXT_OPENID2
```

返回结果如下（关注者列表已返回完时，返回next_openid为空）：

```
{
   "total":23000,
   "count":3000,
   "data":{"
       "openid":[
         "OPENID20001",
         "OPENID20002",
         http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/...,
         "OPENID23000"
       ]
   },
   "next_openid":" "
}
```

4.10 用户分组管理

开发者可以使用接口对公众平台的分组进行查询、创建、修改操作，也可以使用接口在需要时移动用户到某个分组。

4.10.1 创建分组

一个微信公众账号，最多支持创建500个分组。

创建分组的接口如下：

```
https://api.weixin.qq.com/cgi-bin/groups/create?access_token=ACCESS_TOKEN
```

该接口参数说明如表4-28所示。

表4-28　创建分组请求参数说明

参数	说明
access_token	调用接口凭证
name	分组名字（30 个字符以内）

该接口POST数据如下所示：

```
{
    "group": {
        "name": "老师"
    }
}
```

构造一个实例请求如下：

```
https://api.weixin.qq.com/cgi-bin/groups/create?access_token=lQEAXOvehXccOPDfSCCX5_eqV2T47UABa-7e69E0i0egCAOfMle3dQVtHajSOJ66WbFTVBRmFk0RxVYubn6TakyO_uztBpYaeKFKsMPOzwYsoFKLHds
```

创建用户分组的代码实现如下：

```
$access_token = "";
$data = '{"group": {"name": "老师"}}';
$url = "https://api.weixin.qq.com/cgi-bin/groups/create?access_token=$access_token";
$result = https_request($url, $data);
$jsoninfo = json_decode($result, true);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

上述代码执行后，返回结果如下：

```
{
    "group": {
        "id": 100,
        "name": "老师"
    }
}
```

上述结果参数说明如表4-29所示。

表4-29　创建分组结果参数说明

参数	说明
id	分组 id，由微信分配
name	分组名字，UTF8 编码

4.10.2　查询所有分组

查询所有分组的接口如下：

```
https://api.weixin.qq.com/cgi-bin/groups/get?access_token=ACCESS_TOKEN
```

该接口的核心代码实现如下：

```
$url = "https://api.weixin.qq.com/cgi-bin/groups/get?access_token=$access_token";
$result = https_request($url);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

上述代码执行后，返回结果如下：

```
{
    "groups": [
        {
            "id": 0,
            "name": "未分组",
            "count": 2
        },
        {
            "id": 1,
            "name": "黑名单",
            "count": 0
        },
        {
            "id": 2,
            "name": "星标组",
            "count": 0
        },
        {
            "id": 100,
            "name": "老师",
            "count": 0
        }
    ]
```

```
}
```

上述返回结果的参数说明如表4-30所示。

表4-30 查询所有分组结果参数说明

参数	说明	参数	说明
groups	公众平台分组信息列表	name	分组名字，UTF8 编码
id	分组 id，由微信分配	count	分组内用户数量

4.10.3 查询用户所在分组

查询用户所在分组是指通过用户的OpenID查询其所在的GroupID。

查询用户所在分组的接口如下：

```
https://api.weixin.qq.com/cgi-bin/groups/getid?access_token=ACCESS_TOKEN
```

接口运行时，提交POST数据例子如下：

```
{"openid":"o7Lp5t6n59DeX3U0C7Kric9qEx-Q"}
```

请求参数说明如表4-31所示。

表4-31 查询用户所在分组请求参数说明

参数	说明
access_token	调用接口凭证
openid	用户的 OpenID

该接口的核心代码实现如下：

```
$data = '{"openid":"o7Lp5t6n59DeX3U0C7Kric9qEx-Q"}';
$url = "https://api.weixin.qq.com/cgi-bin/groups/getid?access_token=$access_token";
$result = https_request($url, $data);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

上述代码执行后，返回结果如下：

```
{ "groupid": 0}
```

上述返回结果的参数说明如表4-32所示。

表4-32 查询用户所在分组结果参数说明

参数	说明
groupid	用户所属的 GroupID

4.10.4 修改分组名

修改用户所在分组的接口如下：

```
https://api.weixin.qq.com/cgi-bin/groups/update?access_token=ACCESS_TOKEN
```

接口运行时，提交POST数据例子如下：

```
{
"group": {
"id": 100,
"name": "学生"
    }
}
```

该接口的核心代码实现如下：

```
$data = '{"group":{"id":100,"name":"学生"}}';
$url = "https://api.weixin.qq.com/cgi-bin/groups/update?access_token=$access_token";
$result = https_request($url, $data);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

请求参数说明如表4-33所示。

表4-33　修改用户所在分组请求参数说明

参数	说明
access_token	调用接口凭证
id	分组 id，由微信分配
name	分组名字（30 个字符以内）

上述代码执行后，返回结果如下：

```
{"errcode": 0, "errmsg": "ok"}
```

4.10.5　移动用户分组

移动用户所在分组的接口如下：

```
https://api.weixin.qq.com/cgi-bin/groups/members/update?access_token=ACCESS_TOKEN
```

接口运行时，提交POST数据例子如下：

```
{
    "openid": "o7Lp5t6n59DeX3U0C7Kric9qEx-Q",
    "to_groupid": 100
}
```

请求参数说明如表4-34所示。

表4-34　移动用户所在分组请求参数说明

参数	说明
access_token	调用接口凭证
openid	用户唯一标识符
to_groupid	分组 id

该接口的核心代码实现如下：

```
$data = '{"openid":"o7Lp5t6n59DeX3U0C7Kric9qEx-Q","to_groupid":100}';
$url = "https://api.weixin.qq.com/cgi-bin/groups/members/update?access_token=$access_token";
$result = https_request($url, $data);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

上述代码执行后，返回结果如下：

```
{"errcode": 0, "errmsg": "ok"}
```

4.11　上传/下载多媒体文件

微信公众账号在使用接口时，对多媒体文件、多媒体消息的获取和调用等操作，是通过media_id来进行的。通过本接口，微信公众账号可以上传或下载多媒体文件。但需要注意，每个多媒体文件（media_id）会在上传、用户发送到微信服务器3天后自动删除，以节省服务器资源。

4.11.1 上传多媒体文件

微信公众账号可调用本接口来上传图片、语音、视频等文件到微信服务器，上传后服务器会返回对应的media_id，微信公众账号此后可根据该media_id来获取多媒体。请注意，media_id是可复用的，调用该接口需HTTP协议。

通常，文件上传是通过HTML表单进行的，通过CURL可以不经过浏览器，直接在服务器端进行表单的POST提交，完成文件上传功能。

需要注意的是：文件名必须是完整绝对路径，另外需要在绝对路径前加上“@”以示区分。

在Windows服务器上，格式示例为：“@F：\israel\upload\winter.jpg”，而在Linux服务器上，格式示例为：“@home/israel/upload/winter.jpg”。

上传多媒体文件接口如下：

```
http://file.api.weixin.qq.com/cgi-bin/media/upload?access_token=ACCESS_TOKEN&type=TYPE
```

该接口的参数说明如表4-35所示。

表4-35　上传多媒体文件请求参数说明

参数	是否必须	说明
access_token	是	调用接口凭证
type	是	媒体文件类型，分别有图片（image）、语音（voice）、视频（video）和缩略图（thumb）
media	是	form-data 中媒体文件标识，有 filename、filelength、content-type 等信息

1.上传图片

核心代码实现如下：

```
//上传图片文件
$type = "image";
$filepath = dirname(__FILE__)."\winter.jpg";
$filedata = array("media"  => "@".$filepath);
$url = "http://file.api.weixin.qq.com/cgi-bin/media/upload?access_token=$access_token&type=$type";
$result = https_request($url, $filedata);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

返回结果如下所示：

```
{
    "type": "image",
    "media_id": "QQ9nj-7ctrqA8t3WKU3dQN24IuFV_516MfZRZNnQ0c-BFVkk66jUkPXF
49QE9L11",
    "created_at": 1389793969
}
```

2.上传语音

核心代码实现如下：

```
//上传语音
$type = "voice";
$filepath = dirname(__FILE__)."\invite.mp3";
$filedata = array("media"  => "@".$filepath);
$url = "http://file.api.weixin.qq.com/cgi-bin/media/upload?access_token=$access_token&type=$type";
$result = https_request($url, $filedata);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

返回结果如下所示：

```
{
    "type": "voice",
   "media_id": "5Idx79V9E3XfBCz_A50gr1a1_klgPpJnb_eq73yz0bn-prhIsNlwI3n6
jQgshmWk",
    "created_at": 1389794760
}
```

3.上传视频

核心代码实现如下：

```
//上传视频
$type = "video";
$filepath = dirname(__FILE__)."\bbb.mp4";
$filedata = array("media"  => "@".$filepath);
$url = "http://file.api.weixin.qq.com/cgi-bin/media/upload?access_token=$access_token&type=$type";
$result = https_request($url, $filedata);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

返回结果如下所示：

```
{
    "type": "video",
    "media_id": "Jm-Wq0nXtA_oN1qNydQRP03dCsB0R2t5gCHDM3QNkBmMRE1WBaorVJNQTBRHvK9-",
    "created_at": 1389794768
}
```

4.上传缩略图

核心代码实现如下：

```
//上传缩略图
$type = "thumb";
$filepath = dirname(__FILE__)."\qrcode.jpg";
$filedata = array("media"  => "@".$filepath);
$url = "http://file.api.weixin.qq.com/cgi-bin/media/upload?access_token=$access_token&type=$type";
$result = https_request($url, $filedata);
var_dump($result);
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
```

返回结果如下所示：

```
{
    "type": "thumb",
   "thumb_media_id": "2RhP0caRKHVOmZO5AKelHkK--vqPPwgUaRp5-WE63dvmm-
PRWiYVKgvNblIp_gv79",
    "created_at": 1389794771
}
```

以上返回结果的参数说明如表4-36所示。

表4-36　上传多媒体文件结果参数说明

参数	说明
type	媒体文件类型，分别有图片（image）、语音（voice）、视频（video）和缩略图（thumb，主要用于视频与音乐格式的缩略图）
media_id	媒体文件上传后，获取时的唯一标识
created_at	媒体文件上传时间戳

注意事项：上传的多媒体文件有格式和大小限制，具体如下所示。

- 图片（image）：128K，支持JPG格式
- 语音（voice）：256K，播放时间长度不超过60s，支持AMR\MP3格式
- 视频（video）：1MB，支持MP4格式
- 缩略图（thumb）：64KB，支持JPG格式

4.11.2 下载多媒体文件

微信公众账号可调用本接口来获取多媒体文件。请注意，视频文件不支持下载。

下载多媒体文件接口如下：

```
http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=ACCESS_TOKEN&media_id=MEDIA_ID
```

该接口的参数说明如表4-37所示。

表4-37 下载多媒体文件请求参数说明

参数	是否必须	说明
access_token	是	调用接口凭证
media_id	是	媒体文件 ID

1.下载图片

核心代码实现如下：

```
$access_token = "";
//下载图片文件
$mediaid = "QQ9nj-7ctrqA8t3WKU3dQN24IuFV_516MfZRZNnQ0c-BFVkk66jUkPXF49QE9L1l";
$url = "http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=$access_token&media_id=$mediaid";
$fileInfo = downloadWeixinFile($url);
$filename = "down_image.jpg";
saveWeixinFile($filename, $fileInfo["body"]);
function downloadWeixinFile($url)
{
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_NOBODY, 0);    //只取body头
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $package = curl_exec($ch);
    $httpinfo = curl_getinfo($ch);
    curl_close($ch);
    $imageAll = array_merge(array('header' => $httpinfo), array('body' => $package));
    return $imageAll;
}
function saveWeixinFile($filename, $filecontent)
{
    $local_file = fopen($filename, 'w');
    if (false !== $local_file){
        if (false !== fwrite($local_file, $filecontent)) {
            fclose($local_file);
        }
    }
}
```

下载图片时，返回HTTP头示例如下：


```
{
    "url": "http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=
My4oqLEyFVrgFF-XOZagdvbTt9XywYjGwMg_GxkPwql7-f0BpnvXFCOKBUyAf0agmZfMChW5ECSyTAgAoaoU2WMyj7aVHmB17ce4HzLRZ3XFTbm2vpKt_9gYA29xrwIKpnvH-BYmNFSddt7re5ZrIg&media_id=QQ9nj-7ctrqA8t3W
    "content_type": "image/jpeg",
    "http_code": 200,
    "header_size": 308,
    "request_size": 316,
    "filetime": -1,
    "ssl_verify_result": 0,
    "redirect_count": 0,
    "total_time": 1.36,
    "namelookup_time": 1.016,
    "connect_time": 1.078,
    "pretransfer_time": 1.078,
    "size_upload": 0,
    "size_download": 105542,
    "speed_download": 77604,
    "speed_upload": 0,
    "download_content_length": 105542,
    "upload_content_length": 0,
    "starttransfer_time": 1.141,
    "redirect_time": 0
}
```


2.下载语音

核心代码实现如下：

```
//下载语音
$mediaid = "5Idx79V9E3XfBCz_A50gr1a1_klgPpJnb_eq73yz0bn-prhIsNlwI3n6jQgshmWk";
$url = "http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=$access_token&media_id=$mediaid";
$fileInfo = downloadWeixinFile($url);
$filename = "down_voice.mp3";
saveWeixinFile($filename, $fileInfo["body"]);
function downloadWeixinFile($url)
{
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_NOBODY, 0);    //只取body头
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $package = curl_exec($ch);
    $httpinfo = curl_getinfo($ch);
    curl_close($ch);
    $imageAll = array_merge(array('header' => $httpinfo), array('body' =>
$package));
    return $imageAll;
}
function saveWeixinFile($filename, $filecontent)
{
    $local_file = fopen($filename, 'w');
    if (false !== $local_file){
        if (false !== fwrite($local_file, $filecontent)) {
            fclose($local_file);
        }
    }
```

```
}
```

下载语音时，返回HTTP头如下：


```
{
    "url": "http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=
My4oqLEyFVrgFF-XOZagdvbTt9XywYjGwMg_GxkPwql7-f0BpnvXFCOKBUyAf0agmZfMChW5ECSyTAgAoaoU2WMyj7aVHmB17ce4HzLRZ3XFTbm2vpKt_9gYA29xrwIKpnvH-BYmNFSddt7re5ZrIg&media_id=5Idx79V9E3XfBCz_
    "content_type": "audio/amr",
    "http_code": 200,
    "header_size": 306,
    "request_size": 316,
    "filetime": -1,
    "ssl_verify_result": 0,
    "redirect_count": 0,
    "total_time": 0.125,
    "namelookup_time": 0.031,
    "connect_time": 0.063,
    "pretransfer_time": 0.063,
    "size_upload": 0,
    "size_download": 10470,
    "speed_download": 83760,
    "speed_upload": 0,
    "download_content_length": 10470,
    "upload_content_length": 0,
    "starttransfer_time": 0.125,
    "redirect_time": 0
}
```


可以看出，MP3的语音格式被压缩成AMR格式了。

3.下载缩略图

核心代码实现如下：

```
//下载缩略图
$mediaid = "2RhP0caRKHVOmZO5AKelHkK--vqPPwgUaRp5-WE63dvmmPRWiYVKgvNblIp_gv79";
$url = "http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=$access_token&media_id=$mediaid";
$fileInfo = downloadWeixinFile($url);
$filename = "down_thumb.jpg";
saveWeixinFile($filename, $fileInfo["body"]);
function downloadWeixinFile($url)
{
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_NOBODY, 0);    //只取body头
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $package = curl_exec($ch);
    $httpinfo = curl_getinfo($ch);
    curl_close($ch);
    $imageAll = array_merge(array('header' => $httpinfo), array('body' =>
$package));
    return $imageAll;
}
function saveWeixinFile($filename, $filecontent)
{
    $local_file = fopen($filename, 'w');
    if (false !== $local_file){
        if (false !== fwrite($local_file, $filecontent)) {
            fclose($local_file);
        }
    }
}
```

下载缩略图时，返回HTTP头如下：


```
{
    "url": "http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=
My4oqLEyFVrgFF-XOZagdvbTt9XywYjGwMg_GxkPwql7-f0BpnvXFCOKBUyAf0agmZfMChW5ECSyTAgAoaoU2WMyj7aVHmB17ce4HzLRZ3XFTbm2vpKt_9gYA29xrwIKpnvH-BYmNFSddt7re5ZrIg&media_id=2RhP0caRKHVOmZO5
    "content_type": "image/jpeg",
    "http_code": 200,
    "header_size": 306,
    "request_size": 316,
    "filetime": -1,
    "ssl_verify_result": 0,
    "redirect_count": 0,
    "total_time": 0.094,
    "namelookup_time": 0,
    "connect_time": 0.047,
    "pretransfer_time": 0.047,
    "size_upload": 0,
    "size_download": 6057,
    "speed_download": 64436,
    "speed_upload": 0,
    "download_content_length": 6057,
    "upload_content_length": 0,
    "starttransfer_time": 0.094,
    "redirect_time": 0
}
```


4.12 高级群发接口

在公众平台网站上，为订阅号提供了每天一条的群发权限，为服务号提供每月（自然月）4条的群发权限。而对于某些具备开发能力的公众账号运营者，可以通过高级群发接口，实现更灵活的群发能力。

4.12.1 准备群发内容

群发内容可以是文本、图片、语音、视频、图文。群发文本只需要文本内容，其他内容需要获得相应的media_id。

1.文本内容

文本内容就是一段文字，比如：” 微信公众平台开发最佳实践”

2.图片、语音、视频

要求如下：

· 图片（image）：128K，支持JPG格式

· 语音（voice）：256K，播放时间长度不超过60s，支持AMR\MP3格式

· 视频（video）：1MB，支持MP4格式

准备好以后，需要使用上传下载多媒体文件接口将其上传到微信服务器，获得media_id。

3.图文

首先要准备缩略图，要求如下：

· 缩略图（thumb）：64KB，支持JPG格式

同样地，使用上传下载多媒体文件接口，上传到微信服务器后，得到缩略图的media_id，然后需要使用上传图文消息素材接口将其上传到微信服务器，接口为：

```
https://api.weixin.qq.com/cgi-bin/media/uploadnews?access_token=ACCESS_TOKEN
```

要POST提交的数据示例如下：

```
{
    "articles": [
        {
        "thumb_media_id":"qI6_Ze_6PtV7svjolgs-rN6stStuHIjs9_DidOHaj0Q-mwvBelOXCFZiq2OsIU-p",
            "author":"xxx",
            "title":"Happy Day",
            "content_source_url":"www.qq.com",
            "content":"content",
            "digest":"digest"
        },
        {
           "thumb_media_id":"qI6_Ze_6PtV7svjolgs-rN6stStuHIjs9_DidOHaj0Q-mwvBelO
XCFZiq2OsIU-p",
            "author":"xxx",
            "title":"Happy Day",
            "content_source_url":"www.qq.com",
            "content":"content",
            "digest":"digest"
        }
    ]
}
```

参数说明如表4-38所示。

表4-38　群发图文消息参数说明

参数	是否必须	说明
Articles	是	图文消息，一个图文消息支持 1 到 10 条图文
thumb_media_id	是	图文消息缩略图的 media_id，可以在基础支持的上传多媒体文件接口中获得
author	否	图文消息的作者
title	是	图文消息的标题
content_source_url	否	在图文消息页面单击“阅读原文”后的页面
content	是	图文消息页面的内容，支持 HTML 标签
digest	否	图文消息的描述

根据上述POST结构，定义图文数组如下：

```
$news[] = array("thumb_media_id"=>$thumb_media_id1,
                "author"=>"",
                "title"=>"微信公众平台开发",
                "content_source_url" =>"",
                "content" =>"",
                "digest" =>""
                );
$news[] = array("thumb_media_id"=>$thumb_media_id2,
                "author"=>"方倍工作室",
                "title"=>"微信公众平台开发入门教程",
                "content_source_url" =>"http://m.cnblogs.com/99079/3153567.html?full=1",
                "content" =>"<div>
<p>本教程是微信公众平台的入门教程，它将引导你完成如下任务：</p>
<ol>
<li>1. 创建新浪云计算平台应用</li>
<li>2. 启用微信公众平台开发模式</li>
<li>3. 基础接口消息及事件</li>
<li>4. 微信公众平台PHP SDK</li>
<li>5. 微信公众平台开发模式原理</li>
<li>6. 开发天气预报功能</li>
</ol>
</div>",
                "digest" =>"微信公众平台开发经典的入门教程，学习微信公众平台开发必经之路！"
                );
$news[] = array("thumb_media_id"=>$thumb_media_id3,
                "author"=>"方倍工作室",
                "title"=>"微信公众平台开发最佳实践",
                "content_source_url" =>"http://m.cnblogs.com/?u=txw1958",
                "content" =>"<p>本书共分10章，案例程序采用广泛流行的PHP、MySQL、XML、CSS、JavaScript、HTML5等程序语言及数据库实现。系统完整地介绍微信公众平台基础接口、自定义菜单、高级接口、微信
                "digest" =>"微信公众平台开发含金量最高的书籍"
                );
```

将上述图文消息用接口上传成功后，将得到图文消息的media_id数据，返回结果如下所示。

```
{
    "type":"news",
    "media_id":"CsEf3ldqkAYJAU6EJeIkStVDSvffUJ54vqbThMgplD-VJXXof6ctX5fI6-aYyUiQ",
    "created_at":1391857799
```

```
}
```

4.12.2 选择群发对象

群发对象可以是用户组，也可以是OpenID列表。

1.用户组

获得用户分组，需要使用高级接口中的查询所有分组接口，获得相应的group_id，查询到得到用户组数据如下所示。

```
{
    "groups": [
        {
            "id": 0,
            "name": "未分组",
            "count": 2
        },
        {
            "id": 1,
            "name": "黑名单",
            "count": 0
        },
        {
            "id": 2,
            "name": "星标组",
            "count": 0
        },
        {
            "id": 100,
            "name": "老师",
            "count": 0
        }
    ]
}
```

上述数据中的id即用户组id。上述返回结果的参数说明如表4-30所示。

2.OpenID列表

OpenID列表是使用高级接口中的获取关注者列表接口来实现的。获取关注者列表后，得到数据如下所示：

```
{
"total": 200,
"count": 200,
"data": {
"openid": [
"o7Lp5t6n59DeX3U0C7Kric9qEx-Q",
"o7Lp5t5BZDl22PcjIliHp03kzgBE"
        ]
    },
"next_openid": "o7Lp5t5BZDl22PcjIliHp03kzgBE"
}
```

上述返回结果参数说明如表4-27所示。

4.12.3 执行群发

由于群发对象不同，执行群发也有不同的方式。主要是对用户组进行群发和对OpenID列表进行群发。

1.对用户组群发

对用户组群发的接口如下：

```
https://api.weixin.qq.com/cgi-bin/message/mass/sendall?access_token=ACCESS_TOKEN
```

要POST的内容按内容不同组成也不一样。

文本类型的群发数据组成如下所示：

```
{
   "filter":{
      "group_id":"2"
   },
   "text":{
      "content":"CONTENT"
   },
    "msgtype":"text"
}
```

语音类型的群发数据组成如下所示（注意此处media_id需通过基础支持中的上传下载多媒体文件来得到）：

```
{
   "filter":{
      "group_id":"2"
   },
   "voice":{
      "media_id":"123dsdajkasd231jhksad"
   },
    "msgtype":"voice"
}
```

图片类型的群发数据组成如下所示（注意此处media_id需通过基础支持中的上传下载多媒体文件来得到）：

```
{
   "filter":{
      "group_id":"2"
   },
```

```
    "image":{
       "media_id":"123dsdajkasd231jhksad"
    },
     "msgtype":"image"
}
```

视频类型的群发数据组成如下所示：

```
{
    "filter":{
       "group_id":"2"
    },
    "mpvideo":{
       "media_id":"IhdaAQXuvJtGzwwc0abfXnzeezfO0NgPK6AQYShD8RQYMTtfzbLdBIQkQziv2XJc",
    },
     "msgtype":"mpvideo"
}
```

图文消息类型的群发数据组成如下所示（注意图文消息的media_id需要通过上述方法来得到）：

```
{
    "filter":{
       "group_id":"2"
    },
    "mpnews":{
       "media_id":"123dsdajkasd231jhksad"
    },
     "msgtype":"mpnews"
}
```

上述各种类型的相关参数说明如表4-39所示。

表4-39　群发消息类型参数说明

参数	是否必须	说明
filter	是	用于设定图文消息的接收者
group_id	是	群发到的分组的 group_id
mpnews	是	用于设定即将发送的图文消息
media_id	是	用于群发的消息的 media_id
msgtype	是	群发的消息类型，图文消息为 mpnews，文本消息为 text，语音为 voice，音乐为 music，图片为 image，视频为 video
title	否	消息的标题
description	否	消息的描述
thumb_media_id	是	视频缩略图的媒体 id

其接口实现代码如下所示：

```
//高级群发(根据分组)
public function mass_send_group($groupid, $type, $data)
{
        $msg = array('filter' => array('group_id'=>$groupid));
        $msg['msgtype'] = $type;
        switch($type)
        {
                case 'text':
                        $msg[$type] = array('content'=> $data);
                        break;
                case 'image':
                case 'voice':
                case 'mpvideo':
                case 'mpnews':
                        $msg[$type] = array('media_id'=> $data);
                        break;
        }
        $url = "https://api.weixin.qq.com/cgi-bin/message/mass/sendall?access_token=".$this->access_token;
        $res = $this->https_request($url, json_encode($msg));
        return json_decode($res, true);
}
```

上述接口运行后，返回结果如下所示：

```
array(3) {
  ["errcode"]=>
  int(0)
  ["errmsg"]=>
  string(27) "send job submission success"
  ["msg_id"]=>
  float(2347614963)
}
```

上述结果的参数说明如表4-40所示。

表4-40　群发消息结果参数说明

参数	说明
type	媒体文件类型，分别有图片（image）、语音（voice）、视频（video）和缩略图（thumb），图文消息为 news
errcode	错误码
errmsg	错误信息
msg_id	消息 id

图4-20是向用户组发送图文消息后，接收到图文消息和查看图文消息内容后的效果。

图4-20 群发消息成功效果

2.对OpenID列表群发

接口如下：

```
https://api.weixin.qq.com/cgi-bin/message/mass/sendall?access_token=ACCESS_TOKEN
```

POST数据示例如下所示。

文本类型的群发数据如下所示：

```
{
   "touser": [
        "oR5Gjjl_eiZoUpGozMo7dbBJ362A",
        "oR5Gjjo5rXlMUocSEXKT7Q5RQ63Q"
    ],
    "msgtype": "text",
    "text": {
        "content": "hello from boxer."
    }
}
```

语音类型的群发数据如下所示：

```
{
   "touser":[
        "OPENID1",
        "OPENID2"
```

```
    ],
    "voice":{
        "media_id":"mLxl6paC7z2Tl-NJT64yzJve8T9c8u9K2x-Ai6Ujd4lIH9IBuF6-2r66mamn_gIT"
    },
    "msgtype":"voice"
}
```

图片类型的群发数据如下所示：

```
{
    "touser":[
     "OPENID1",
     "OPENID2"
    ],
    "image":{
        "media_id":"BTgN0opcW3Y5zV_ZebbsD3NFKRWf6cb7OPswPi9Q83fOJHK2P67dzxn11Cp7THat"
    },
    "msgtype":"image"
}
```

视频类型的群发数据如下所示：

```
{
    "filter":{
        "group_id":"2"
    },
    "mpvideo":{
        "media_id":"IhdaAQXuvJtGzwwc0abfXnzeezfO0NgPK6AQYShD8RQYMTtfzbLdBIQkQziv2XJc",
    },
     "msgtype":"video"
}
```

图文消息类型的群发数据如下所示（注意图文消息的media_id需要通过上述方法来得到）：

```
{
    "touser":[
         "OPENID1",
         "OPENID2"
    ],
    "mpnews":{
         "media_id":"123dsdajkasd231jhksad"
    },
    "msgtype":"mpnews"
}
```

上述消息的参数列表如表4-41所示。

表4-41　群发OpenID列表消息参数说明

参数	是否必须	说明
touser	是	填写图文消息的接收者，一串 OpenID 列表，OpenID 最少 1 个，最多 10 000 个
mpnews	是	用于设定即将发送的图文消息
media_id	是	用于群发的图文消息的 media_id
msgtype	是	群发的消息类型，图文消息为 mpnews，文本消息为 text，语音为 voice，音乐为 music，图片为 image，视频为 video
title	否	消息的标题
description	否	消息的描述
thumb_media_id	是	视频缩略图的媒体 id

返回数据示例（正确时的JSON返回结果）：

```
{
     "errcode":0,
     "errmsg":"send job submission success",
     "msg_id":2347614964
}
```

4.12.4　接收群发结果

群发任务提交后，群发任务可能在一定时间后才完成，因此，群发接口调用时，仅会给出群发任务是否提交成功的提示，若群发任务提交成功，则在群发任务结束时，向开发者在公众平台填写的开发者URL（callback URL）推送事件。

推送的XML结构如下（发送成功时）：

```
<xml>
     <ToUserName><![CDATA[gh_3e8adccde292]]></ToUserName>
     <FromUserName><![CDATA[oR5Gjjl_eiZoUpGozMo7dbBJ362A]]></FromUserName>
     <CreateTime>1394524295</CreateTime>
     <MsgType><![CDATA[event]]></MsgType>
     <Event><![CDATA[MASSSENDJOBFINISH]]></Event>
     <MsgID>1988</MsgID>
     <Status><![CDATA[sendsuccess]]></Status>
     <TotalCount>100</TotalCount>
     <FilterCount>80</FilterCount>
     <SentCount>75</SentCount>
     <ErrorCount>5</ErrorCount>
</xml>
```

上述XML结构的参数如表4-42所示。

表4-42　群发消息结果内容参数说明

参数	说明
ToUserName	公众账号的微信号
FromUserName	公众账号群发助手的微信号，为 mphelper
CreateTime	创建时间的时间戳
MsgType	消息类型，此处为 event
Event	事件信息，此处为 MASSSENDJOBFINISH
MsgID	群发的消息 id
Status	群发的结构，为“send success”或“send fail”或“err(num)”。但 send success 时，也有可能因用户拒收公众账号的消息、系统错误等原因造成少量用户接收失败
TotalCount	group_id 下粉丝数；或者 openid_list 中的粉丝数
FilterCount	过滤（过滤是指特定地区、性别的过滤、用户设置拒收的过滤，用户接收已超 4 条的过滤）后，准备发送的粉丝数，原则上，FilterCount = SentCount + ErrorCount
SentCount	发送成功的粉丝数
ErrorCount	发送失败的粉丝数

在微信公众平台PHP SDK中增加该事件消息的处理结果如下：

```
private function receiveEvent($object)
{
    $content = "";
    switch ($object->Event)
    {
        case "subscribe":
            $content = "欢迎关注方倍工作室";
            break;
        case "MASSSENDJOBFINISH":
            $content = "消息ID：".$object->MsgID.
                       "\n结果：".$object->Status.
                       "\n粉丝数：".$object->TotalCount.
                       "\n过滤：".$object->FilterCount.
                       "\n发送成功：".$object->SentCount.
                       "\n发送失败：".$object->ErrorCount;
            break;
        default:
            break;
    }
    $result = $this->transmitText($object, $content);
    return $result;
}
```

4.13　模板消息

模板消息仅用于公众账号向用户发送重要的服务通知，且只能用于符合其要求的服务场景中，如信用卡刷卡通知、商品购买成功通知等。

4.13.1　添加模板

使用模板消息需要先添加模板到模板库中，在微信公众平台后台依次选择“模板消息”→“模板库”，可以看到自己所申请的行业的模板列表，如图4-21所示。

图4-21　模板库

选择要添加的模板右侧的“详情”链接，可以看到模板的详细信息，如图4-22所示。

图4-22　模板详情

单击底部的“添加”按钮，该模板将被添加到模板库中，如图4-23所示。

图4-23　我的模板

单击已添加模板旁的“详情”链接，可以看到该模板ID和该模板消息的详细内容及参数，如图4-24所示。

图4-24　模板ID和详细参数

4.13.2　模板消息开发

根据模板消息ID和详细内容中的参数，我们构造模板消息包的JSON格式如下。

```
{
    "touser": "owddJuAiiQpXZedAWxjpp3pkZTzU",
    "template_id": "jD1Jfu0ElKcyEK0CfJ2JjTy4U1fjYI09l6eax9BBu9U",
    "url": "http://www.cnblogs.com/txw1958/",
    "topcolor": "#7B68EE",
    "data": {
        "first": {
            "value": "您好，方倍，欢迎使用模板消息！",
            "color": "#743A3A"
        },
        "product": {
            "value": "微信公众平台开发最佳实践",
            "color": "#FF0000"
        },
        "price": {
            "value": "69.00元",
            "color": "#C4C400"
        },
        "time": {
            "value": "2014年6月1日",
            "color": "#0000FF"
        },
        "remark": {
            "value": "\n你的订单已提交，我们将尽快发货。祝您生活愉快！",
            "color": "#008000"
        }
    }
}
```

发送模板消息的实现代码如下：

```
<?php
$template = array('touser' => "owddJuAiiQpXZedAWxjpp3pkZTzU",
                  'template_id' => "jD1Jfu0ElKcyEK0CfJ2JjTy4U1fjYI09l6eax9BBu9U",
                  'url' => "http://www.cnblogs.com/txw1958/",
                  'topcolor' => "#7B68EE",
                  'data' => array('first'    => array('value' => urlencode("您好，方倍，欢迎使用模板消息！"),
                                                       'color' => "#743A3A",
                                                       ),
                                  'product' => array('value' => urlencode("微信公众平台开发最佳实践"),
                                                       'color' => "#FF0000",
                                                       ),
                                  'price'     => array('value' => urlencode("69.00元"),
                                                       'color' => "#C4C400",
                                                       ),
                                  'time'     => array('value' => urlencode("2014年6月1日"),
                                                       'color' => "#0000FF",
                                                       ),
                                  'remark'     => array('value' => urlencode("\\n你的订单已提交，我们将尽快发货。祝您生活愉快！"),
                                                       'color' => "#008000",
                                                       ),
                                  )
                  );
$url = "https://api.weixin.qq.com/cgi-bin/message/template/send?access_token=".$access_token;
$result = https_request($url, urldecode(json_encode($template)));
var_dump($result);
function https_request($url,$data = null){
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}
?>
```

发送成功后，效果如图4-25所示。

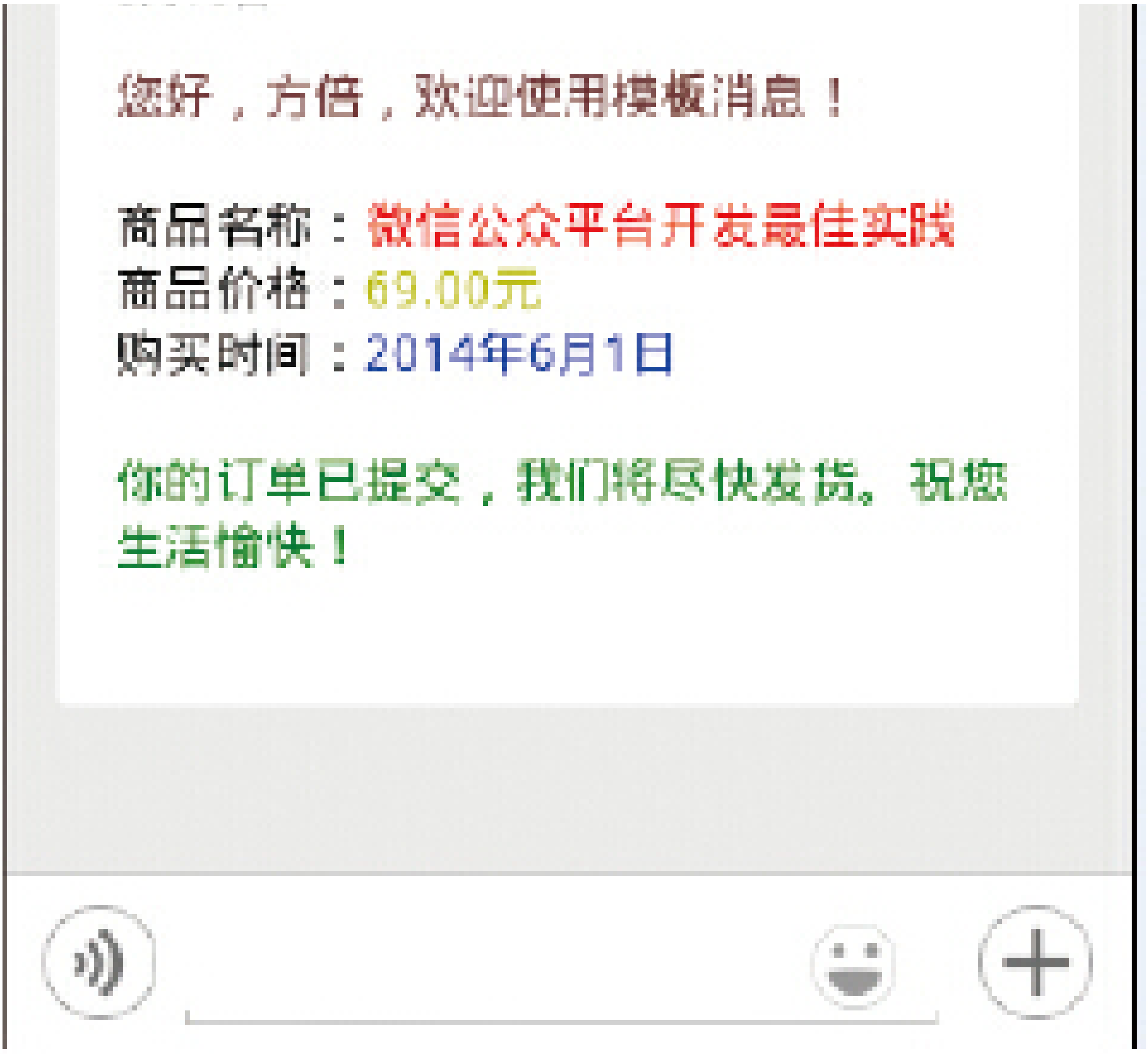


图4-25　模板消息

4.14　本章小结

本章介绍了微信公众平台主要高级接口的原理及实现方法，这些接口包括自定义菜单、语音识别、客服接口、OAuth2.0网页授权、带参数的二维码、获取用户地理位置、获取用户基本信息、获取关注者列表、用户分组管理、上传下载多媒体文件、高级群发接口、模板消息等。这些接口可以帮助开发出更灵活、更强大的应用。

第5章　微信支付

微信支付是由腾讯微信与财付通联合推出的移动支付产品，旨在为广大微信用户及商户提供优质安全的支付服务，其支付和安全系统由持有互联网支付牌照并具备完备的安全体系的第三方支付平台财付通提供支持。

5.1　申请微信支付

5.1.1　申请流程图

申请成为微信支付商家，必须满足三个条件：

1）拥有微信公众平台（http://mp.weixin.qq.com）公众账号，且为服务号。

2）公众账号必须经过微信认证。

3）目前支持企业、媒体、网店商家类型认证的服务号。

通过微信认证的服务号，进入申请页面http://mp.weixin.qq.com，登录后，单击“服务”功能栏下的“服务中心”模块，可看到右侧的“商户功能”模块，如图5-1所示。

图5-1　微信公众平台商户功能

商户功能，是微信公众平台向有出售物品需求的公众账号提供推广销售、支付收款、经营分析的整套解决方案。

商户通过自定义菜单、关键字回复等方式向订阅用户推送商品消息，用户可在微信公众账号中完成选购支付的流程。

商户也可以把商品网页转换成二维码，张贴在线下的场景，如车站和广告海报。用户扫描后可查看商品详情，在微信中直接购买。

商户功能申请界面如图5-2所示。

图5-2　商户功能申请界面

商户功能是微信支付申请的入口。微信支付的申请流程图如图5-3所示。

图5-3　微信支付申请流程图

申请流程详细说明如下。

第一步：完成服务号认证。

（1）微信支付功能目前仅对完成微信认证的服务号开放申请（企业、媒体、政府及其他组织）。若公众账号符合开放申请要求，可直接进入第二步（微信公众平台提交资料）。

（2）订阅号可先升级为服务号，升级方法：登录“微信公众平台”→“设置”→“账号信息”→“升级为服务号”。

（3）未认证的服务号需先完成微信认证。

注意，商户申请微信认证的主体与申请开通微信支付功能的主体需保持一致。

第二步：完成申请资料审核。

（1）登录“微信公众平台”，进入：“服务”→“服务中心”→“商户功能”。

（2）提交商户基本资料，提交时准确选择经营范围，并如实填写出售的商品/服务信息，此处填写的信息将作为日后运营监管的依据。

需要准备的材料包括（扫描件请务必加盖公司公章）：

1）140字以内的出售商品描述。

2）企业客服电话。

3）支付授权目录、维权通知URL、支付回调URL、告警通知URL（由商户开发人员提供）。

4）在open.weixin.qq.com注册得到的APPID（由商户开发人员提供）。

（3）提交业务审核资料。商户提交的资料，主体需与微信认证主体保持一致，以保证运营主体即认证主体。

需要准备的材料包括（扫描件请务必加盖公司公章）：

1）公司微信支付负责人姓名、联系电话、邮箱

2）公司网站首页链接（非必备）

3）商家名称（将用于对外展示）

4）公司地址及邮政编码

5）营业执照持有人类型、姓名、所持证件及扫描件

6）税务登记证扫描件

（4）提交财务审核资料。商户提交的财务资料，主体需与业务审核资料主体一致，以保证结算主体即运营主体；商户提交的所有资料需加盖公章。

需要准备的材料包括（扫描件请务必加盖公司公章）：

1）公司对公账户名称

2）公司对公账户账号

3）开户行名称

4）开户行所在地

（5）资料审核。

1）商户申请资料提交成功后，腾讯在7个工作日内反馈审核结果。

2）审核结果将以电子邮件的形式告知商户。商户也可登录“微信公众平台”，单击页面右上角小信封图标查看。

3）审核通过的通知邮件中将包含非常重要的开发参数，请商户牢记申请时填写的“重要邮箱”地址，相关通知一经发送至“重要邮箱”地址，则视为腾讯已经向商户履行了通知义务。“重要邮箱”是商户在填写“业务审核资料”时设置的，其填写界面如图5-4所示。

第三步：功能开发、合同签订。

（1）通过资料审核的商户可以进行功能开发工作：腾讯提供清晰的开发接口文档，帮助商户顺利完成功能开发工作。为了不耽误进入申请流程中的第四步，即“缴纳保证金”、“申请发布全网“及”销售商品“流程，建议商户将合同签订与功能开发同步进行。

图5-4　重要邮箱

（2）签订合同。

1）资料审核通过后，商户对合同进行盖章确认后请根据要求寄至腾讯。

2）若申请开通公众账号支付功能，商户需签订《微信公众平台商户功能服务协议》和《微信支付服务协议》。

3）若申请开通APP支付功能，商户只需签订《微信支付服务协议》，而无需签订《微信公众平台商户功能服务协议》。

4）若申请同时开通公众账号支付和APP支付功能，需签订《微信公众平台商户功能服务协议》和《微信支付服务协议》。

5）腾讯在收到商户寄回的合同后，会由专人负责审核，确认无误后，会尽快盖章并按照约定的份数寄还商户。（注：商户盖章时请注意加盖骑缝章。）

第四步：开通商户功能。

（1）缴纳风险保证金：

1）商户登录财付通账户缴纳风险保证金；该财付通账户的登录id和密码可在资料审核通过时腾讯发送的通知邮件中查看，登录id即partnerid，登录密码即partnerkey。

2）未缴纳风险保证金不会影响开通商户功能，但保证金足额缴纳是款项结算服务开通的前提。

（2）开通商户功能是指测试白名单之外的微信号也能在商户的公众账号内使用微信支付功能，开通商户功能之后才能在公众账号内出售商品或服务。

5.1　申请微信支付

5.1.1　申请流程图

申请成为微信支付商家，必须满足三个条件：

1）拥有微信公众平台（http://mp.weixin.qq.com）公众账号，且为服务号。

2）公众账号必须经过微信认证。

3）目前支持企业、媒体、网店商家类型认证的服务号。

通过微信认证的服务号，进入申请页面http://mp.weixin.qq.com，登录后，单击“服务”功能栏下的“服务中心”模块，可看到右侧的“商户功能”模块，如图5-1所示。

图5-1　微信公众平台商户功能

商户功能，是微信公众平台向有出售物品需求的公众账号提供推广销售、支付收款、经营分析的整套解决方案。

商户通过自定义菜单、关键字回复等方式向订阅用户推送商品消息，用户可在微信公众账号中完成选购支付的流程。

商户也可以把商品网页转换成二维码，张贴在线下的场景，如车站和广告海报。用户扫描后可查看商品详情，在微信中直接购买。

商户功能申请界面如图5-2所示。

图5-2　商户功能申请界面

商户功能是微信支付申请的入口。微信支付的申请流程图如图5-3所示。

图5-3　微信支付申请流程图

申请流程详细说明如下。

第一步：完成服务号认证。

（1）微信支付功能目前仅对完成微信认证的服务号开放申请（企业、媒体、政府及其他组织）。若公众账号符合开放申请要求，可直接进入第二步（微信公众平台提交资料）。

（2）订阅号可先升级为服务号，升级方法：登录“微信公众平台”→“设置”→“账号信息”→“升级为服务号”。

（3）未认证的服务号需先完成微信认证。

注意，商户申请微信认证的主体与申请开通微信支付功能的主体需保持一致。

第二步：完成申请资料审核。

（1）登录“微信公众平台”，进入：“服务”→“服务中心”→“商户功能”。

（2）提交商户基本资料，提交时准确选择经营范围，并如实填写出售的商品/服务信息，此处填写的信息将作为日后运营监管的依据。

需要准备的材料包括（扫描件请务必加盖公司公章）：

1）140字以内的出售商品描述。

2）企业客服电话。

3）支付授权目录、维权通知URL、支付回调URL、告警通知URL（由商户开发人员提供）。

4）在open.weixin.qq.com注册得到的APPID（由商户开发人员提供）。

（3）提交业务审核资料。商户提交的资料，主体需与微信认证主体保持一致，以保证运营主体即认证主体。

需要准备的材料包括（扫描件请务必加盖公司公章）：

1）公司微信支付负责人姓名、联系电话、邮箱

2）公司网站首页链接（非必备）

3）商家名称（将用于对外展示）

4）公司地址及邮政编码

5）营业执照持有人类型、姓名、所持证件及扫描件

6）税务登记证扫描件

（4）提交财务审核资料。商户提交的财务资料，主体需与业务审核资料主体一致，以保证结算主体即运营主体；商户提交的所有资料需加盖公章。

需要准备的材料包括（扫描件请务必加盖公司公章）：

1）公司对公账户名称

2）公司对公账户账号

3）开户行名称

4）开户行所在地

（5）资料审核。

1）商户申请资料提交成功后，腾讯在7个工作日内反馈审核结果。

2）审核结果将以电子邮件的形式告知商户。商户也可登录“微信公众平台”，单击页面右上角小信封图标查看。

3）审核通过的通知邮件中将包含非常重要的开发参数，请商户牢记申请时填写的“重要邮箱”地址，相关通知一经发送至“重要邮箱”地址，则视为腾讯已经向商户履行了通知义务。“重要邮箱”是商户在填写“业务审核资料”时设置的，其填写界面如图5-4所示。

第三步：功能开发、合同签订。

（1）通过资料审核的商户可以进行功能开发工作：腾讯提供清晰的开发接口文档，帮助商户顺利完成功能开发工作。为了不耽误进入申请流程中的第四步，即“缴纳保证金”、“申请发布全网“及”销售商品“流程，建议商户将合同签订与功能开发同步进行。

图5-4　重要邮箱

（2）签订合同。

1）资料审核通过后，商户对合同进行盖章确认后请根据要求寄至腾讯。

2）若申请开通公众账号支付功能，商户需签订《微信公众平台商户功能服务协议》和《微信支付服务协议》。

3）若申请开通APP支付功能，商户只需签订《微信支付服务协议》，而无需签订《微信公众平台商户功能服务协议》。

4）若申请同时开通公众账号支付和APP支付功能，需签订《微信公众平台商户功能服务协议》和《微信支付服务协议》。

5）腾讯在收到商户寄回的合同后，会由专人负责审核，确认无误后，会尽快盖章并按照约定的份数寄还商户。（注：商户盖章时请注意加盖骑缝章。）

第四步：开通商户功能。

（1）缴纳风险保证金：

1）商户登录财付通账户缴纳风险保证金；该财付通账户的登录id和密码可在资料审核通过时腾讯发送的通知邮件中查看，登录id即partnerid，登录密码即partnerkey。

2）未缴纳风险保证金不会影响开通商户功能，但保证金足额缴纳是款项结算服务开通的前提。

（2）开通商户功能是指测试白名单之外的微信号也能在商户的公众账号内使用微信支付功能，开通商户功能之后才能在公众账号内出售商品或服务。

5.1.2　开放经营类目

首期开放微信支付的经营类目及相应资质要求如表5-1所示。

表5-1　首期经营类目

一级类目	二级类目	三级类目	商品类型	资质要求
实物	综合商城	线下商超		《企业法人营业执照》、店铺门面照和经营场所照
	服饰鞋包	服饰		《企业法人营业执照》
		鞋类箱包		《企业法人营业执照》
	运动户外	运动户外		《企业法人营业执照》
	美妆用品	化妆品 / 个人护理	国产	《化妆品生产企业卫生许可证》或《化妆品生产许可证》(化妆品生产行政许可与化妆品卫生行政许可两项行政许可整合为一项行政许可)
			进口	报关单复印件、检验检疫证明，以及非特殊用途化妆品备案凭证附件或特殊用途化妆品许可批件复印件
	家装家纺	家装建材		《企业法人营业执照》
		居家家纺		《企业法人营业执照》
		鲜花礼品		《企业法人营业执照》
	汽车及配件	整车预订 / 汽配		《企业法人营业执照》
	数码家电	3C 数码		《企业法人营业执照》
		家用电器		《企业法人营业执照》
	母婴	奶粉	国产	《食品流通许可证》或《食品卫生许可证》
			进口	《食品流通许可证》或《食品卫生许可证》，以及《出入境检验检疫卫生证书》
		玩具		《全国工业产品生产许可证》
		母婴服饰和用品		《企业经营许可证》
	食品	实体店餐饮		《餐饮服务许可证》、店铺门面照和经营场所照
		普通食品	国产	《食品流通许可证》或《食品卫生许可证》
			进口	《食品流通许可证》或《食品卫生许可证》，以及《出入境检验检疫卫生证书》
		酒类	进口	《食品流通许可证》、《酒类流通备案登记表》、《进口食品卫生证》和《进口食品标签审核证》
			生产	《食品流通许可证》和《酒类流通备案登记表》
			批发	《食品流通许可证》和《酒类流通备案登记表》
			零售	《食品流通许可证》和《酒类流通备案登记表》
	图书	图书		《出版许可证》或出版社授权文件，以及《出版物经营许可证》
		音像		《音像制品出版许可证》或出版社授权文件，以及《音像制品经营许可证》
		报纸		《报纸出版许可证》或报纸出版授权文件，以及《出版物经营许可证》
		杂志		《杂志出版许可证》或杂志社授权文件，以及《出版物经营许可证》

5.1.3 资费标准

商户使用商户功能和微信支付功能时，需按照协议规定的标准缴纳支付商户功能服务费及微信支付手续费，在商户与腾讯签署的《微信公众平台商户功能服务协议》及《微信支付服务协议》中已对费率及收费方式做了具体约定。

为了更好地保障消费者权益、规范商户管理，使用微信支付功能的商户需要按照合同约定缴纳一定金额的风险保证金。商户将通过其财付通账户缴纳风险保证金并冻结在商户财付通账户中，保证金的具体使用或解冻以相关规则或协议要求为准。

首期开放微信支付的经营类目及相应风险保证金、微信支付手续费费率、交易结算周期如表5-2所示。

表5-2　保证金、费率、结算周期

一级类目	二级类目	三级类目	商品类型	保证金	费率	结算周期
实物	综合商城	线下商超		20 000	0.60%	T+7
	服饰鞋包	服饰		20 000	0.60%	T+7
		鞋类箱包		20 000	0.60%	T+7
	运动户外	运动户外		20 000	0.60%	T+7
	美妆用品	化妆品 / 个人护理	国产	20 000	0.60%	T+7
			进口	20 000	0.60%	T+7
	家装家纺	家装建材		20 000	0.60%	T+7
		居家家纺		20 000	0.60%	T+7
		鲜花礼品		20 000	0.60%	T+7
	汽车及配件	整车预订 / 汽配		20 000	0.60%	T+7
	数码家电	3C 数码		20 000	0.60%	T+7
		家用电器		20 000	0.60%	T+7
	母婴	奶粉	国产	20 000	0.60%	T+7
			进口	20 000	0.60%	T+7
		玩具		20 000	0.60%	T+7
		母婴服饰和用品		20 000	0.60%	T+7
	食品	实体店餐饮		20 000	0.60%	T+7
		普通食品	国产	20 000	0.60%	T+7
			进口	20 000	0.60%	T+7
		酒类	进口	20 000	0.60%	T+7
			生产	20 000	0.60%	T+7
			批发	20 000	0.60%	T+7
			零售	20 000	0.60%	T+7
	图书	图书		20 000	0.60%	T+7
		音像		20 000	0.60%	T+7
		报纸		20 000	0.60%	T+7
		杂志		20 000	0.60%	T+7

5.1.4 常见问题

（1）经营类目中找不到我公司出售的商品/服务，怎么办?

为了更好地提供商户功能服务，加强对交易风险的控制和管理，首期仅开放部分经营类目申请微信支付功能，后续将陆续开放更多经营类目，敬请期待。

（2）提交完申请资料之后一直没有通过审核，为什么?

收到商户申请资料后，腾讯将在7个工作日内反馈审核结果。若申请被驳回，商户可登录微信公众平台查看驳回原因。请尽快根据驳回原因调整申请内容或资质文件，并再次提交，腾讯将再次审核。

（3）个人可以申请微信支付吗?

微信支付功能暂不接受个人申请。

（4）商户缴纳的微信风险保证金、商户功能服务费，能否开具发票或收据?

腾讯会为商户开具商户功能使用服务费的发票。不开具风险保证金发票，因风险保证金冻结于商户财付通账户中，腾讯并未收取此款项，所以腾讯不提供风险保证金发票或收据。

（5）没有公众账号，如何申请APP支付权限?

近期将开放APP微信支付功能申请。届时，没有开通微信公众账号的APP开发者可在微信开放平台进行申请。

5.2 微信支付功能介绍

5.2.1 方案简介

微信支付提供体验顺畅、使用安全的移动支付解决方案。它以微信支付为核心功能，提供微信地址共享、支付结算、客户关系维护、售后维权、交易统计的整套移动购物解决方案。其方案包括的内容如图5-5所示。

图5-5　微信支付方案

5.2　微信支付功能介绍

5.2.1　方案简介

微信支付提供体验顺畅、使用安全的移动支付解决方案。它以微信支付为核心功能，提供微信地址共享、支付结算、客户关系维护、售后维权、交易统计的整套移动购物解决方案。其方案包括的内容如图5-5所示。

图5-5　微信支付方案

5.2.2　应用场景举例

商户通过图文消息、自定义菜单、关键字回复等方式向订阅用户推送商品消息，用户可在微信公众账号中完成选购、填写收货地址、支付的流程。

商户也可以把商品网页转换成二维码，发布在线下和线上的媒体，如车站、楼宇广告以及网页广告。用户用微信“扫一扫”功能后可打开商品详情，在微信中直接购买。

1.网页内购买场景——JS API支付接口

（1）微信公众账号店铺

在微信公众账号内放置商品信息入口，利用自定义菜单、关键字回复和CallBack接口下发商品信息，用户可在微信公众账号中完成商品选购、填写收货地址、支付的流程。通过微信OpenID可实现用户免注册、免登录的效果。购买全流程如图5-6所示。

图5-6　微信公众账号店铺购买全流程

上述流程解析如下：在微信公众账号中设置店铺菜单；单击菜单后，跳转到商城页面，系统使用OAuth2.0接口自动获得用户基本信息，其中包括用户微信OpenID身份，无需用户手动登录；用户单击购买时，调用收货地址共享接口获得用户已经填写的地址列表，省去用户填写地址的时间，提高转化率；用户单击“微信支付”按钮时，调用微信支付接口，引导用户输入支付密码，完成支付。

（2）微信扫购模式

相比于微信公众账号店铺，微信扫购模式能够帮助商户更方便地自主营销，受众群体也不局限于公众账号的关注用户，也可拓展到线上、线下更为广大的用户群体。

商户可把商品信息或二维码发布到线上、线下，每一个平面都可以成为商户的销售渠道。

用户扫描二维码后，弹出订单确认页面，页面再引导用户完成支付。流程如图5-7所示。

图5-7　微信扫购流程图

2.扫码支付场景——Native（原生）支付接口

与网页内支付场景不同，部分用户不需要经过网页选购，可以直接扫码支付购买。

商户根据微信支付的规则，为不同商品生成不同的二维码，张贴在各种场景，便于用户扫描购买。用户使用微信扫描二维码后获取商品信息，同时到商户后台下单。然后用户开始支付，输入支付密码。支付成功后，商户后台得到通知，进行发货处理。流程如图5-8所示。

图5-8　微信扫码支付流程图

3.收货地址共享

收货地址共享，是指可在微信中调用微信的地址组件，首次调用此地址需用户授权，用户授权后，收货地址可在不同网页中共享使用，地址数据会传递到商户后台。

新用户进入网页，没有获取到收货地址。用户单击“新增收货地址”，进入微信地址控件填写地址，然后商户可获得此数据。

对于已有地址的用户，进入网页时可以快速选择已有地址，完成购买流程，而且地址数据作为用户属性会长久保存。用户单击收货地址，可以新增、删除或者编辑地址。这里的所有流程在客户端界面完成，数据会同步到云端。

流程如图5-9所示。

图5-9　微信收货地址

4.接入维权系统

用户在公众账号内交易出现纠纷时，通常会联系腾讯客服，因此腾讯需要即时了解公众账号与用户交易的详情。同时，为了最快效率地解决用户的问题，腾讯作为连接用户和商户的桥梁，会通过此维权系统即时将问题同步给商户，并将解决结果反馈至用户。

所有获得微信支付功能的公众账号必须接入用户维权系统。用户添加维权界面如图5-10所示。

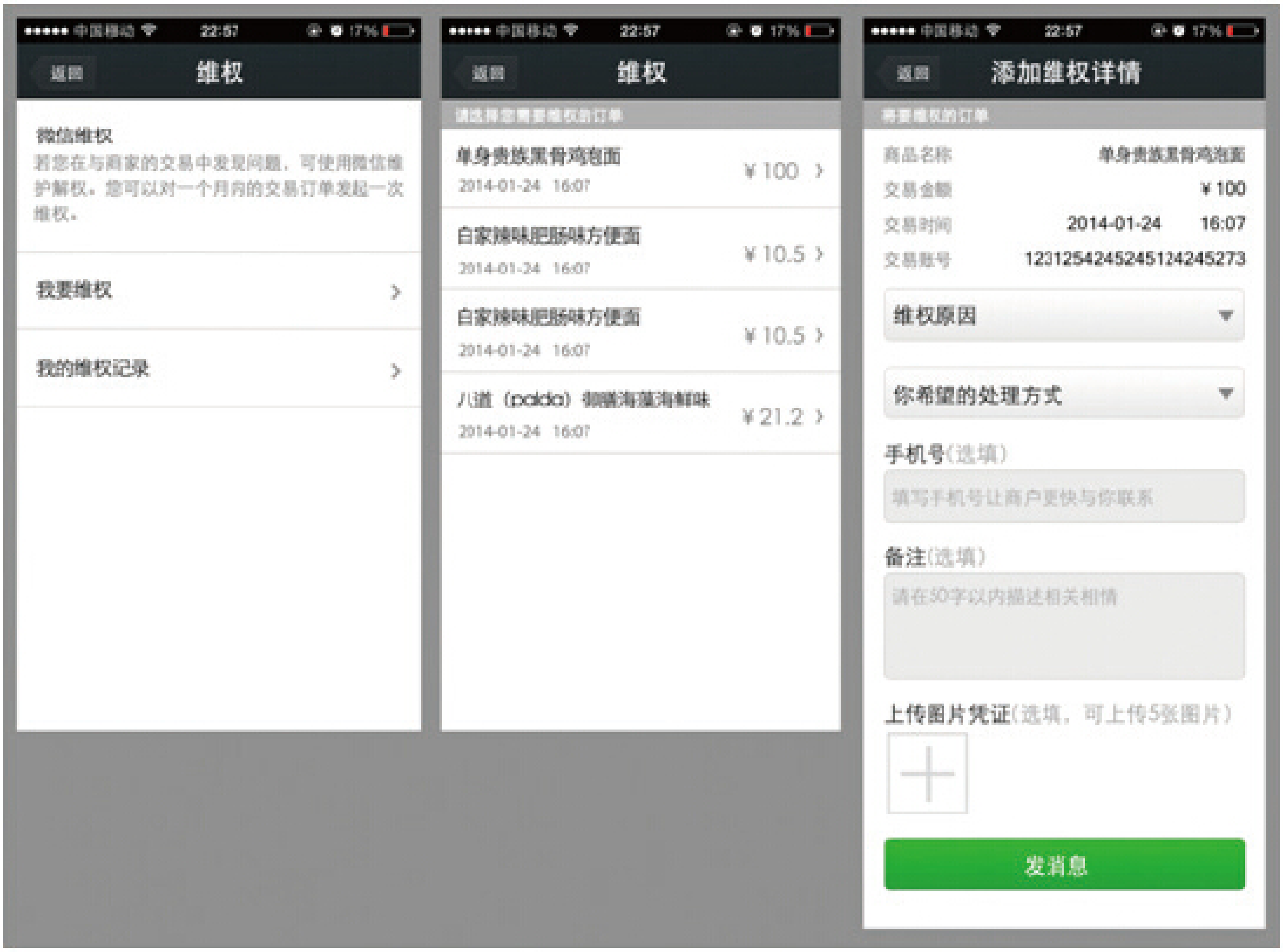


图5-10　微信维权

查看维权状态界面如图5-11所示。

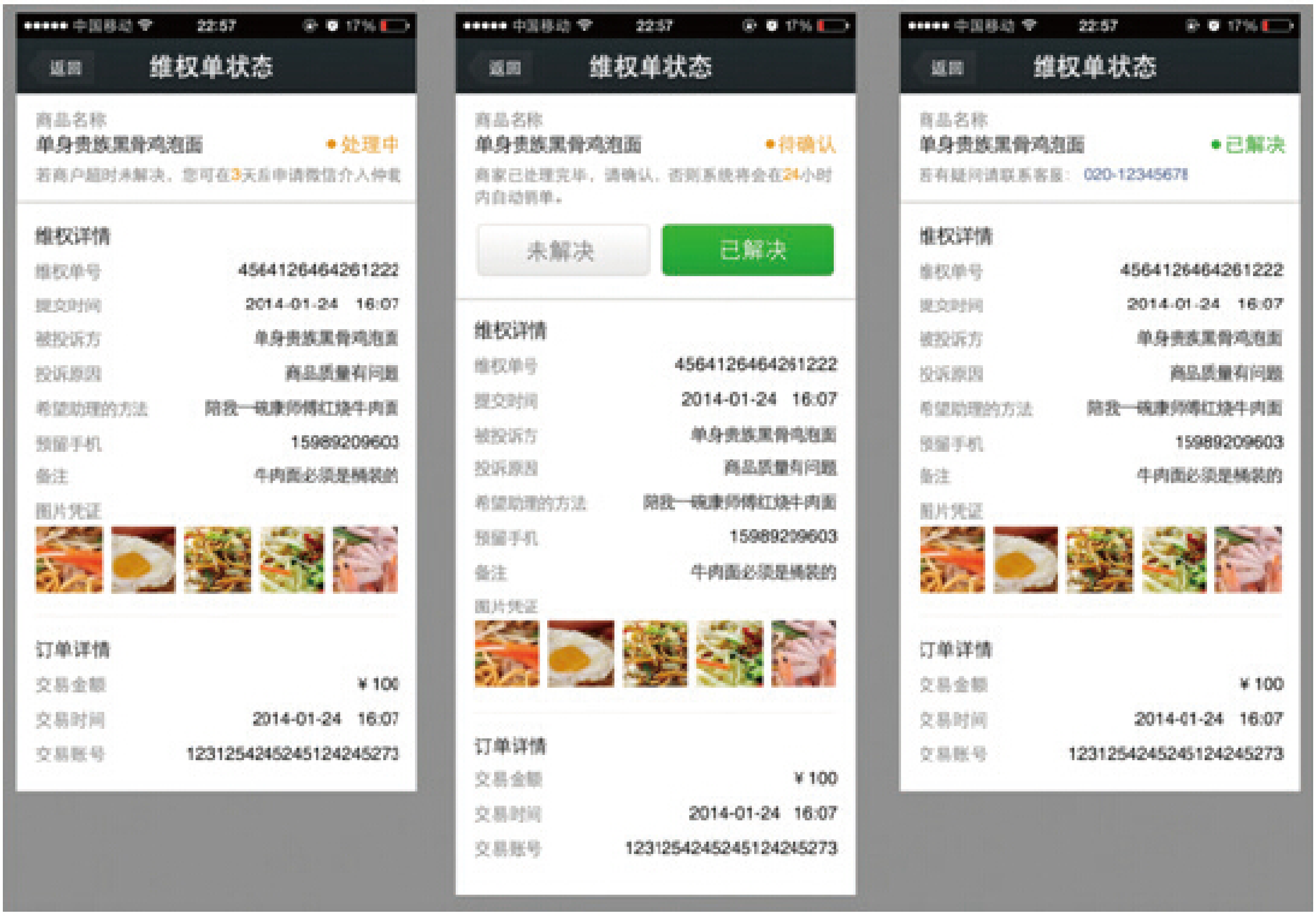


图5-11　维权单状态

5.APP支付

微信APP支付，可以让手机APP直接调用微信支付，整个过程非常顺畅，可有效提高购买转化率，便于用户重复购买。

微信APP支付界面如图5-12所示。

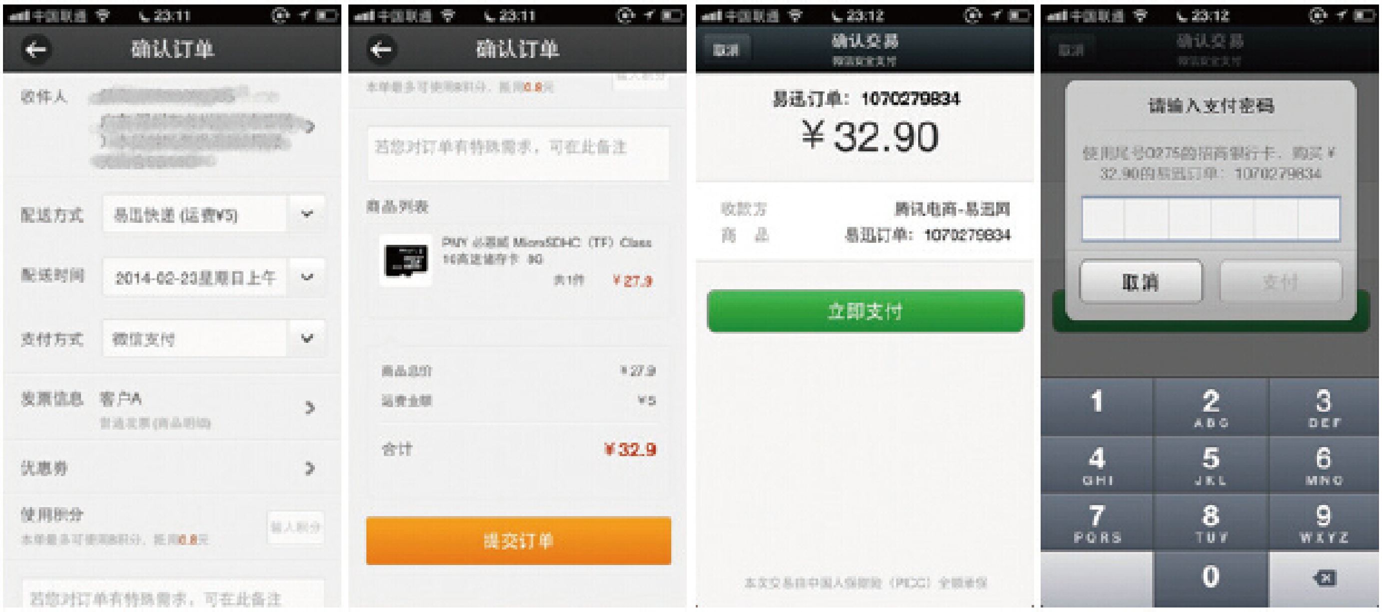


图5-12　微信APP支付

申请时需要在公众平台勾选APP支付，并填写第三方应用的APPID。微信APP支付后期也会在微信开放平台上开放申请入口。

5.2.3　商户模块

开通微信支付功能后，可以登录公众平台使用商户模块功能，查看交易相关的统计数据，也可使用财付通的商户号，登录财付通企业版，查看订单流水，进行退款、下载对账单等操作。

商品信息，是特殊类型的图文消息，用户打开后直接跳转到商户网站的商品详情页，而不经过公众平台中间页。商户可以通过群发、实时回复、自动回复等方式，把商品信息下发给用户。群发商品信息界面如图5-13所示。

图5-13　群发商品信息

订单流水是汇总了一定周期内商户交易的流水情况，数据包括订单信息、金额、买家、渠道来源等。商户也可以从订单中直接向买家发起对话。订单流水信息界面如图5-14所示。

图5-14　订单流水

开通商户功能后，会为商户分配一个财付通商户号（PartnerID），可以登录财付通企业版管理资金和进行退款、下载对账单等操作。财付通账户资金信息界面如图5-15所示。

图5-15　财付通账户信息

商户在获得支付权限后处于支付测试状态，只有白名单内的用户可以在微信客户端上使用支付功能，此时需要在“支付测试”模块中添加白名单。在各项准备工作完成后，商户可发起发布申请，申请通过后支付功能才能被全部用户使用。

5.2.4 功能接口

表5-3是商户可能用到的功能和接口，具体可根据商户的业务定制开通。

表5-3 商户可能用到的功能和接口

编号	类型	功能描述	开通条件
1	用户身份识别	即 OAuth2.0 用户授权接口，商户可通过该接口得到每一个用户加密且唯一的 OpenID，获取用户的相关信息，进而识别用户身份，记录用户消费行为，提供更好的服务	完成微信认证
2	支付功能	选择以下至少一种的支付方式 1）JS API 支付：通过浏览器单击或者扫描二维码打开的商品详情网页，选购商品后，调用客户端的微信支付功能； 2）Native（原生）支付：可直接通过扫描二维码，进入原生购买页面，适用没有选购流程的支付场景	完成三项支付资料审核
3	支付相关接口	提供支付订单查询、退款、下载对账单等接口，辅助商户完成支付流程；同时，商户必须接入发货通知接口，每笔交易均同步发货状态到微信	完成三项支付资料审核
4	共享收货地址	用户在下单网页内填写的地址将保存到微信支付平台，商户可共享用户在微信原生控件中填写过的地址，实现自动填写，可增加和编辑	基本资料填写勾选且完成三项支付资料审核

（续）

编号	类型	功能描述	开通条件
5	购买后发货	用户下单后，商户必须在规定时间内发货，并使用发货通知接口，将发货信息传递给微信平台。如果发货超时，将依据平台运营规则进行警告或处罚	完成三项支付资料审核
6	售后维权	处理用户投诉，有效地解决用户购买后发货掉单的投诉问题，微信会监控商户的经营情况。商户必须接入维权系统，才能提供上线支付等相关服务	完成三项支付资料审核
7	告警通知	若检测到商户经营行为异常或服务不稳定，微信平台将通过告警接口将相关信息告知到商户。未来将可能包括发货超时告警、服务器异常告警、支付存在风险告警等	完成三项支付资料审核

5.2.5 账号体系

商户在微信公众平台提交申请资料以及银行账户资料，资料审核通过并签约后，可以获得表5-4中所示账户（包含财付通的相关支付资金账户），用于公众账号支付。

表5-4 微信支付账号体系

账号	作用
appId	公众账号身份的唯一标识。审核通过后，在微信发送的邮件中查看
appSecret	公众账号支付请求中用于加密的密钥 Key，可验证商户唯一身份，PaySignKey 对应于支付场景中的 appKey 值。审核通过后，在微信发送的邮件中查看
paySignKey	除了支付请求需要用到 paySignKey，公众平台接口 API 的权限获取所需密钥 Key，在使用所有公众平台 API 时都需要先用它去换取 access_token，然后再进行调用。审核通过后，在微信发送的邮件中查看
partnerId	财付通商户身份的标识。审核通过后，在财付通发送的邮件中查看
partnerKey	财付通商户权限密钥 Key。审核通过后，在财付通发送的邮件中查看

appSecret、paySignKey、partnerKey是验证商户唯一性的安全标识，需妥善保管。对于appSecret和paySignKey的区别，可以这样认为：appSecret是API使用时的登录密码，会在网络中传播；而paySignKey是在所有支付相关数据传输时用于加密并进行身份校验的密钥，仅保留在第三方后台和微信平台，不会在网络中传播，而且paySignKey仅用于支付请求。

5.2.6 商户系统对接

开通商户功能后，可按以下步骤完成商户和微信平台的对接。

1）生成商品消息：通过接口或者登录公众平台网站，根据商户网站的商品网页生成对应的消息，也可根据网页链接生成二维码。

2）商品推广：通过群发、自定义菜单、关键字回复等方式下发商品消息，也可把二维码张贴在车站、海报等线下的场景。

3）用户浏览：用户可通过接收消息或者扫描二维码打开商品网页，浏览商品详情。

4）用户下单：选购商品，可确认款式、数量和收货地址等信息，确认下单。

5）用户支付：商户调用微信支付功能发起交易；用户确认交易信息，验证支付密码即可。

6）支付成功：用户看到支付成功的结果，通知商户后台，可更新订单状态、进行发货等相应操作。

7）发货通知：商户收到支付成功通知，发货后需要通知微信平台，并由微信记录状态。

8）客户维权：客服系统即时将问题同步给商户，商户可以联系用户解决。

9）资金结算：按照约定的结算周期，系统结算资金到商户的银行账户，同时根据合同的费率收取支付服务费。

10）退款操作：商户使用分配的财付通商户号，登录财付通企业版（http://mch.tenpay.com/）手动操作退款，也可以使用后台接口进行退款。

11）下载对账单：商户除了登录财付通企业版手动下载订单流水，也可以使用后台接口下载对账单。

5.2.7 行业案例

依托微信庞大的用户群、极致的用户体验，知名B2C电商纷纷对接微信支付，积极参与构建移动电商生态环境。图5-16、图5-17分别显示了1号店和酒仙网的微信支付案例。

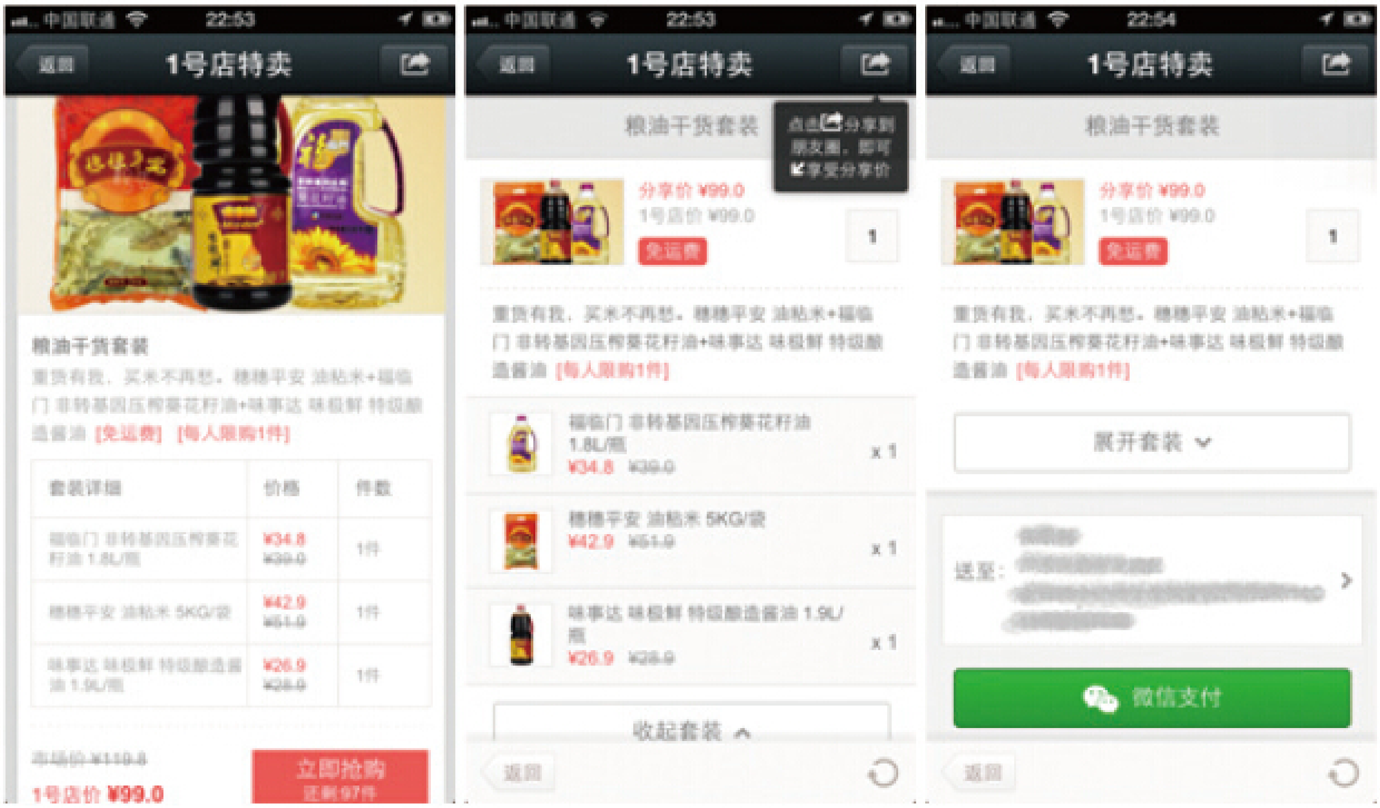


图5-16　1号店

结合传统的售卖机，用户选购商品后，扫描机器上的二维码，使用Native（原生）支付进行快速购买。微信可提供支付账号和硬件芯片的解决方案。图5-18显示了友宝的微信原生支付案例。

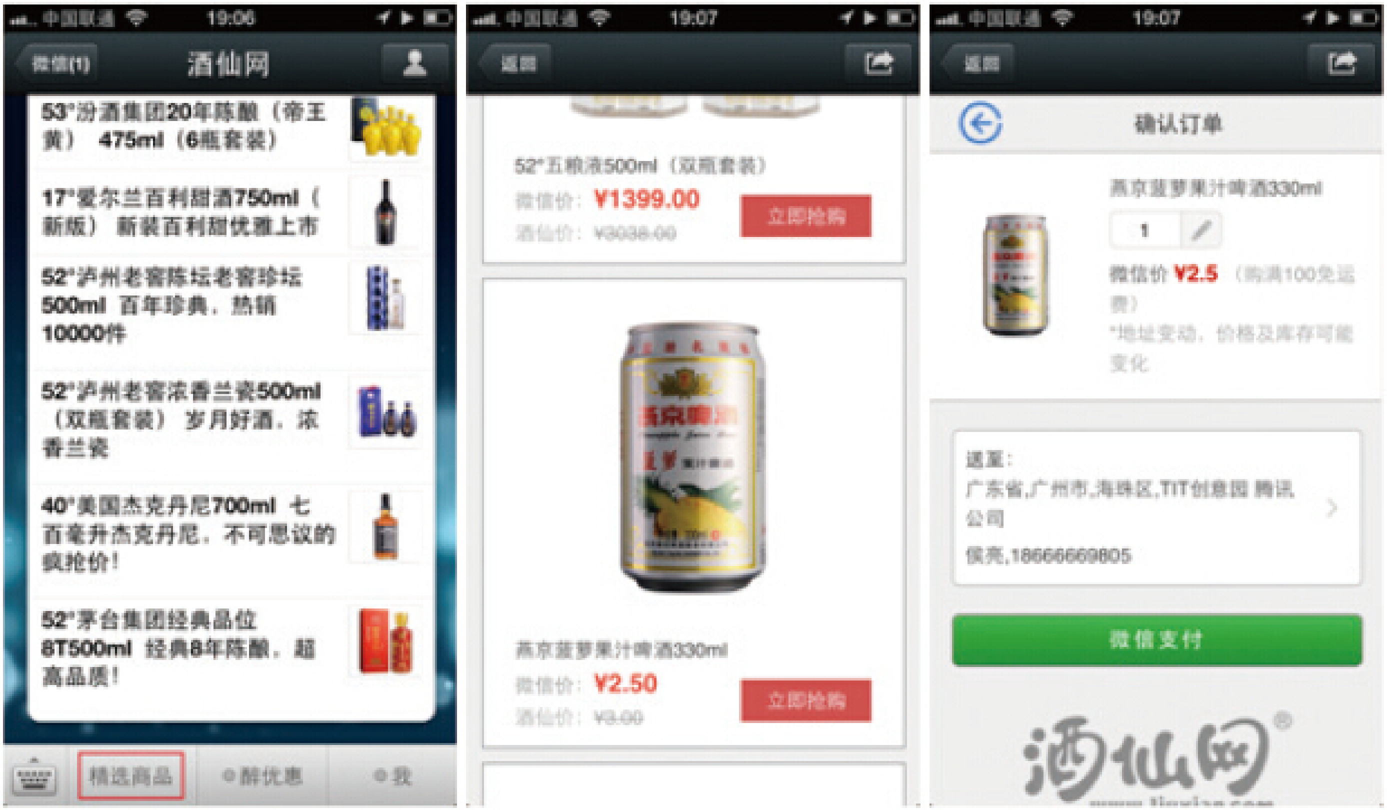

中国联通
19:06
微信(1)
酒仙网
53°汾酒集团20年陈酿（帝王黄） 475ml（6瓶套装）
17°爱尔兰百利甜酒750ml（新版） 新装百利甜优雅上市
52°泸州老窖陈坛老窖珍坛500ml 百年珍典，热销10000件
52°泸州老窖浓香兰瓷500ml（双瓶套装） 岁月好酒，浓香兰瓷
40°美国杰克丹尼700ml 七百毫升杰克丹尼，不可思议的疯抢价！
52°茅台集团经典品位8T500ml 经典8年陈酿，超高品质！
精选商品
醉优惠
我
中国联通
19:07
返回
52°五粮液500ml（双瓶套装）
微信价：¥1399.00
酒仙价：¥3038.00
立即抢购
燕京菠萝果汁啤酒330ml
微信价：¥2.50
酒仙价：¥3.00
立即抢购
中国联通
19:07
返回
确认订单
燕京菠萝果汁啤酒330ml
1
微信价 ¥2.5
送至：
广东省,广州市,海珠区,TIT创意园 腾讯公司
微信支付
酒仙网

图5-17　酒仙网

图5-18　友宝

5.2.8　支付功能常见问题

（1）微信支付和财付通的关系?

微信支付是以绑定银行卡为基础的支付服务，由财付通提供支付牌照和银行通道的技术支持。用户使用微信支付绑定银行卡，和微信号对应QQ使用财付通的绑定无关。同时，用户使用微信号对应QQ登录财付通，无法查看微信支付绑定的银行卡和对应的交易记录。

对于用户，主要认知的是微信支付品牌。对于商户，除了拥有微信公众账号的商户身份，还会有微信公众平台分配的一个财付通商户号，用于结账收款。

（2）微信支付如何绑定银行卡？如何支付?

用户首次绑定银行卡时，需输入银行卡号和持卡人信息并验证银行预留手机号。用户还需设置支付密码（非银行取款密码），后续购买不用再绑卡，验证支付密码即可直接支付。微信支付的绑定和支付流程，都是标准的快捷支付流程。

（3）微信支付可以绑定的银行卡数量?

微信支付支持绑定13家银行，22个卡种的银行卡，包括储蓄卡和信用卡都可以。目前正在不断接入银行，覆盖范围会越来越大。

（4）微信除了提供商户的支付接口，还有其他接口能力吗?

除了支付接口，微信还会提供订单查询、退款、退款查询、对账单下载等支付相关的接口，和财付通提供给商户的接口能力一样。不少商户担心退款和对账的接口，这些都是可以提供的。具体参考《财付通支付网关商户开发指南》。

（5）微信支付的支付类型是什么?

微信支付使用的是即时到账接口，暂不支持担保交易类型交易。

（6）收货地址共享的地址格式是?

微信收货地址是基于三级的国标数据，地区对应是国标三级地区码，如“广东省-广州市-天河区”，对应的地区码是510630。

（7）微信支付的安全性如何保证?

有以下手段保证支付安全:

1）绑定银行卡需要验证实名信息和银行预留手机号；绑定银行卡后设置支付密码，交易需要验证此支付密码，大额支付还需要验证手机短信。

2）微信认证商户网页是安全的支付环境，有微信安全支付的认证标题提示用户。

3）支付后台有腾讯的大数据支撑，海量的数据和云端的计算能够及时判定用户的支付行为存在的风险性，如异地支付、非常用设备支付、QQ号被盗等情况，分析和拦截异常支付行为。

4）7*24小时客户服务，加上微信客服，将及时为用户排忧解难；同时为微信支付开辟专属客服通道，将以最快的速度响应用户提出的问题并做出处理判断。

（8）接入微信支付，商户担心网页会被假冒和钓鱼?

微信会认证安全商户的网页，提示“微信安全支付”；不安全或者高风险的网页，微信会弹出中间页提示不安全并拦截，未受安全认证的网页内填写相关账号或者支付，系统也会提示“网页内提示不要输入QQ密码或请勿支付”。

（9）商户接入微信支付的成本高吗?

对于已有H5商城的商户接入JS API支付，前端页面上加“微信支付”按钮即可，支付成功后，后台接收通知发货；若接入Native（原生）支付，商户后台增加一个下单接口，接收微信的下单请求即可。开发成本较低，不需要改变已有的流程。

（10）商户使用微信支付需要有一个财付通商户号，能否使用已有的商户号?

由于支付权限和支付费率的差异，不能使用已有的财付通商户号，需要在微信公众平台重新申请开通。

5.3　微信支付接口

微信支付接口包含一系列的功能及接口，主要有支付接口、收货地址接口、用户维权接口、通知接口。另外还有APP支付接口，用于微信APP上的支付。

5.3.1　支付基础

由于微信5.0版本后才加入微信支付模块，低版本用户调用微信支付功能将无效。因此需要通过user agent来确定用户当前的版本号后再调用支付接口。以iPhone上的微信为例，可以通过user agent获取如下微信版本示例信息:

```
Mozilla/5.0(iphone;CPU iphone OS 5_1_1 like Mac OS X) AppleWebKit/534.46(KHTML,like Geocko) Mobile/9B206 MicroMessenger/5.0
```

其中5.0为用户安装的微信版本号，商户可以判定版本号是否高于或者等于5.0。

对于具有支付权限且需要调用微信支付的商户页面，为了让用户增加购买信心，确认交易环境安全，可以使用“微信安全支付”标题。安全支付标题的图5-19所示。

图5-19　微信安全支付

为了显示支付安全标题，需在原始链接添加上“showwxpaytitle=1”的尾串。通过这种方式，商户的页面才将出现微信安全支付的标识。例如，原始URL为：http://weixin.qq.com，显示安全支付标题的URL为：http://weixin.qq.com?showwxpaytitle=1。当用户在微信里打开http://weixin.qq.com不会直接出现微信安全支付的标题，而打开http://weixin.qq.com?showwxpaytitle=1后将出现微信安全支付标题。

下面是微信支付定义的一些基础类及其方法，分别介绍如下。

1.公共函数类

公共函数类用于支付接口中一些常用功能的定义，文件为CommonUtil.php，相关代码及注释如下所示:

```
class CommonUtil
{
    //连接url与参数
    function genAllUrl($toURL, $paras)
    {
        $allUrl = null;
        if(null == $toURL){
            die("toURL is null");
        }
        if (strripos($toURL,"?") =="") {
            $allUrl = $toURL . "?" . $paras;
        }else {
```

```
            $allUrl = $toURL . "&" . $paras;
        }
        return $allUrl;
    }

    //生成随机16位字符串
    function create_noncestr( $length = 16 ) {
        $chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
        $str ="";
        for ( $i = 0; $i < $length; $i++ )  {
            $str.= substr($chars, mt_rand(0, strlen($chars)-1), 1);
        }
        return $str;
    }

    //没有用到该函数
    function splitParaStr($src, $token)
    {
        $resMap = array();
        $items = explode($token,$src);
        foreach ($items as $item){
            $paraAndValue = explode("=",$item);
            if ($paraAndValue != "") {
                $resMap[$paraAndValue[0]] = $parameterValue[1];
            }
        }
        return $resMap;
    }

    //去除空白字符串
    function trimString($value)
    {
        $ret = null;
        if (null != $value) {
            $ret = $value;
            if (strlen($ret) == 0) {
                $ret = null;
            }
        }
        return $ret;
    }

    //将参数字典序排序，使用URL键值对的格式拼接成字符串
    function formatQueryParaMap($paraMap, $urlencode)
    {
        $buff = "";
        ksort($paraMap);
        foreach ($paraMap as $k => $v){
            if (null != $v && "null" != $v && "sign" != $k) {
                if($urlencode){
                   $v = urlencode($v);
                }
                $buff .= $k . "=" . $v . "&";
            }
        }
        $reqPar;
        if (strlen($buff) > 0) {
            $reqPar = substr($buff, 0, strlen($buff)-1);
        }
        return $reqPar;
    }

    //将参数字典序排序，使用URL键值对的格式拼接成字符串，字符串转小写
    function formatBizQueryParaMap($paraMap, $urlencode)
    {
        $buff = "";
        ksort($paraMap);
        foreach ($paraMap as $k => $v){
            if($urlencode){
               $v = urlencode($v);
            }
            $buff .= strtolower($k) . "=" . $v . "&";
        }
        $reqPar;
        if (strlen($buff) > 0) {
            $reqPar = substr($buff, 0, strlen($buff)-1);
        }
        return $reqPar;
    }

    //数组转xml
    function arrayToXml($arr)
    {
        $xml = "<xml>";
        foreach ($arr as $key=>$val)
        {
             if (is_numeric($val)){
                 $xml.="<".$key.">".$val."</".$key.">";
             }else{
                 $xml.="<".$key."><![CDATA[".$val."]]></".$key.">";
             }
        }
        $xml.="</xml>";
        return $xml;
    }
}
```

2.MD5签名类

MD5签名类用于生成MD5签名及校验签名，文件为MD5SignUtil.php，相关代码及注释如下所示：

```
class MD5SignUtil
{
    //连接参数，md5签名后再转成大写
    function sign($content, $key)
    {
        try {
            if (null == $key) {
               throw new SDKRuntimeException("财付通签名key不能为空！" . "<br>");
            }
            if (null == $content) {
               throw new SDKRuntimeException("财付通签名内容不能为空" . "<br>");
            }
            $signStr = $content . "&key=" . $key;
            return strtoupper(md5($signStr));
        }catch (SDKRuntimeException $e)
        {
            die($e->errorMessage());
        }
    }

    //验证md5签名
    function verifySignature($content, $sign, $md5Key)
    {
        $signStr = $content . "&key=" . $md5Key;
        $calculateSign = strtolower(md5($signStr));
        $tenpaySign = strtolower($sign);
        return $calculateSign == $tenpaySign;
    }
}
```

3.异常处理类

异常处理类用于返回异常，文件为SDKRuntimeException.class.php，相关代码及注释如下所示：

```
class SDKRuntimeException extends Exception
{
    //返回错误
    public function errorMessage()
    {
        return $this->getMessage();
    }
}
```

4.配置文件

配置文件定义账号等配置信息，文件为WxPay.config.php，相关代码及注释如下所示：

```
define(APPID , "wxf8b4f85f3a794e77");                   //appid
define(APPKEY ,"2Wozy2aksie1puXUBpWD8oZxiD1DfQuEaiC7KcRATv1Ino3mdopKaPGQQ7TtkNySuAmCaDCrw4xhPY5qKTBl7Fzm0RgR3c0WaVYIXZARsxzHV2x7iwPPzOz94dnwPWSn");   //paysign key
define(SIGNTYPE, "sha1");                              //加密方法
define(PARTNERKEY,"8934e7d15453e97507ef794cf7b0519d");        //通加密串
define(APPSERCERT, "09cb46090e586c724d52f7ec9e60c9f8");
```

5.微信支付类

微信支付类定义微信支付过程中的函数及方法，文件为WxPayHelper.php，相关代码及注释如下所示：

```
class WxPayHelper
{
    var $parameters; //配置参数

    //构造函数
    function __construct()
    {

    }

    //设置参数
    function setParameter($parameter, $parameterValue)
    {
        $this->parameters[CommonUtil::trimString($parameter)] = CommonUtil::trimString($parameterValue);
    }

    //获取参数
    function getParameter($parameter)
    {
        return $this->parameters[$parameter];
    }

    //生成16位随机字符串
    protected function create_noncestr( $length = 16 )
    {
        $chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
        $str ="";
        for ( $i = 0; $i < $length; $i++ )  {
            $str.= substr($chars, mt_rand(0, strlen($chars)-1), 1);
        }
        return $str;
    }

    //检查配置参数
    function check_cft_parameters()
    {
        if($this->parameters["bank_type"] == null || $this->parameters["body"] == null || $this->parameters["partner"] == null ||
            $this->parameters["out_trade_no"] == null || $this->parameters["total_fee"] == null || $this->parameters["fee_type"] == null ||
            $this->parameters["notify_url"] == null || $this->parameters["spbill_create_ip"] == null || $this->parameters["input_charset"] == null
            ){
            return false;
        }
        return true;
    }

    //生成package订单详情
    protected function get_cft_package()
    {
        try {
            if (null == PARTNERKEY || "" == PARTNERKEY ) {
                throw new SDKRuntimeException("密钥不能为空！" . "<br>");
            }
            $commonUtil = new CommonUtil();
            ksort($this->parameters);
            $unSignParaString = $commonUtil->formatQueryParaMap($this->parameters, false);
            $paraString = $commonUtil->formatQueryParaMap($this->parameters, true);
            $md5SignUtil = new MD5SignUtil();
            return $paraString."&sign=".$md5SignUtil->sign($unSignParaString,$commonUtil->trimString(PARTNERKEY));
        }catch (SDKRuntimeException $e){
            die($e->errorMessage());
        }
    }

    //生成签名
    protected function get_biz_sign($bizObj)
    {
        foreach ($bizObj as $k => $v){
             $bizParameters[strtolower($k)] = $v;
        }
        try {
            if(APPKEY == ""){
                throw new SDKRuntimeException("APPKEY为空！" . "<br>");
            }
            $bizParameters["appkey"] = APPKEY;
            ksort($bizParameters);
            $commonUtil = new CommonUtil();
            $bizString = $commonUtil->formatBizQueryParaMap($bizParameters, false);
            return sha1($bizString);
        }catch (SDKRuntimeException $e){
            die($e->errorMessage());
        }
    }

    //生成app支付请求json
    function create_app_package($traceid="")
    {
        try {
           if($this->check_cft_parameters() == false) {
                throw new SDKRuntimeException("生成package参数缺失！" . "<br>");
            }
            $nativeObj["appid"] = APPID;
            $nativeObj["package"] = $this->get_cft_package();
            $nativeObj["timestamp"] = time();
            $nativeObj["traceid"] = $traceid;
            $nativeObj["noncestr"] = $this->create_noncestr();
            $nativeObj["app_signature"] = $this->get_biz_sign($nativeObj);
            $nativeObj["sign_method"] = SIGNTYPE;
            return json_encode($nativeObj);
        }catch (SDKRuntimeException $e){
            die($e->errorMessage());
```

```
        }
    }

    //生成jsapi支付请求json
    function create_biz_package()
    {
        try {
            if($this->check_cft_parameters() == false) {
               throw new SDKRuntimeException("生成package参数缺失！" . "<br>");
            }
           $nativeObj["appId"] = APPID;
           $nativeObj["package"] = $this->get_cft_package();
           $nativeObj["timeStamp"] = strval(time());
           $nativeObj["nonceStr"] = $this->create_noncestr();
           $nativeObj["paySign"] = $this->get_biz_sign($nativeObj);
           $nativeObj["signType"] = SIGNTYPE;
           return   json_encode($nativeObj);
        }catch (SDKRuntimeException $e)
        {
            die($e->errorMessage());
        }
    }

    //生成原生支付url
    function create_native_url($productid)
    {
        $commonUtil = new CommonUtil();
        $nativeObj["appid"] = APPID;
        $nativeObj["productid"] = urlencode($productid);
        $nativeObj["timestamp"] = time();
        $nativeObj["noncestr"] = $this->create_noncestr();
        $nativeObj["sign"] = $this->get_biz_sign($nativeObj);
        $bizString = $commonUtil->formatBizQueryParaMap($nativeObj, false);
        return "weixin://wxpay/bizpayurl?".$bizString;
    }

    //生成原生支付请求xml
    function create_native_package($retcode = 0, $reterrmsg = "ok")
    {
        try {
            if($this->check_cft_parameters() == false && $retcode == 0) {
                throw new SDKRuntimeException("生成package参数缺失！" . "<br>");
            }
            $nativeObj["AppId"] = APPID;
            $nativeObj["Package"] = $this->get_cft_package();
            $nativeObj["TimeStamp"] = time();
            $nativeObj["NonceStr"] = $this->create_noncestr();
            $nativeObj["RetCode"] = $retcode;
            $nativeObj["RetErrMsg"] = $reterrmsg;
            $nativeObj["AppSignature"] = $this->get_biz_sign($nativeObj);
            $nativeObj["SignMethod"] = SIGNTYPE;
            $commonUtil = new CommonUtil();
            return  $commonUtil->arrayToXml($nativeObj);
        }catch (SDKRuntimeException $e)
        {
            die($e->errorMessage());
        }
    }
}
```

5.3.2 JS API支付接口

1.JS API支付接口（getBrandWCPayRequest）

微信JS API只能在微信内置浏览器中使用，其他浏览器调用则无效。微信提供getBrand-WCPayRequest接口供商户前端网页调用，调用之前微信会鉴定商户支付权限，若商户具有调用支付的权限，则将开始支付流程。这里主要介绍支付前的接口调用规则，支付状态消息通知机制请参加下文。需要注意：所有传入参数都是字符串类型！

getBrandWCPayRequest参数如表5-5所示。

表5-5　getBrandWCPayRequest参数

参数	名称	必填	格式	说明
appId	公众账号 id	是	字符串类型	商户注册具有支付权限的公众账号成功后即可获得
timeStamp	时间戳	是	字符串类型，32 个字节以下	商户生成，从 1970 年 1 月 1 日 00:00:00 至今的秒数，即当前的时间，且最终需要转换为字符串形式
nonceStr	随机字符串	是	字符串类型，32 个字节以下	商户生成的随机字符串
package	订单详情扩展字符串	是	字符串类型，4096 个字节以下	商户将订单信息组成该字符串，具体组成方案参见接口使用说明中 package 组包帮劣；由商户按照规范拼接后传入
signType	签名方式	是	字符串类型，参数取值 "SHA1"	按照文档中所示填入，目前仅支持 SHA1
paySign	签名	是	字符串类型	商户将接口列表中的参数按照指定方式进行签名，签名方式使用 signType 中标示的签名方式，具体签名方案参见接口使用说明中签名帮助；由商户按照规范签名后传入

getBrandWCPayRequest返回值如表5-6所示。

表5-6　getBrandWCPayRequest返回值

返回值	说明
err_msg	get_brand_wcpay_request:ok 支付成功 get_brand_wcpay_request:cancel 支付过程中用户取消 get_brand_wcpay_request:fail 支付失败

JS API的返回结果get_brand_wcpay_request：ok仅在用户成功完成支付时返回。由于前端交互复杂，get_brand_wcpay_request：cancel或者get_brand_wcpay_request：fail可以统一处理为用户遇到问题或者主动放弃，不必细化区分。

JS API支付接口使用示例如下：

```
include_once("WxPayHelper.php");

$commonUtil = new CommonUtil();
$wxPayHelper = new WxPayHelper();
$wxPayHelper->setParameter("bank_type", "WX");
$wxPayHelper->setParameter("body", "test");
$wxPayHelper->setParameter("partner", "1900000109");
$wxPayHelper->setParameter("out_trade_no", $commonUtil->create_noncestr());
$wxPayHelper->setParameter("total_fee", "1");
$wxPayHelper->setParameter("fee_type", "1");
$wxPayHelper->setParameter("notify_url", "htttp://www.baidu.com");
$wxPayHelper->setParameter("spbill_create_ip", "127.0.0.1");
$wxPayHelper->setParameter("input_charset", "GBK");
?>
<html>
<script language="javascript">
    function callpay()
    {
        WeixinJSBridge.invoke('getBrandWCPayRequest',<?php echo $wxPayHelper->create_biz_package(); ?>,function(res){
            WeixinJSBridge.log(res.err_msg);
            alert(res.err_code+res.err_desc+res.err_msg);
        });
    }
</script>
<body>
    <button type="button" onclick="callpay()">wx pay test</button>
</body>
</html>
```

其在后台实际传输的值如下面代码所示。

```
//传入公众账号名称，时间戳，随机串，Package 扩展字段，签名方式和 PaySign 签名
WeixinJSBridge.invoke(
    'getBrandWCPayRequest',{
        "appId" : "wxf8b4f85f3a794e77",
        "timeStamp" : "189026618",
        "nonceStr" : "adssdasssd13d",
        "package" : "bank_type=WX&body=XXX&fee_type=1&input_charset=GBK&notify_url=http%3a%2f%2fwww.qq.com&out_trade_no=16642817866003386000&partner=1900000109&spbill_create
        "signType" : "SHA1",
        "paySign" : "7717231c335a05165b1874658306fa431fe9a0de"
    },function(res){
        //  返回 res.err_msg,取值
        // get_brand_wcpay_request:cancel  用户取消
        // get_brand_wcpay_request:fail  发送失败
        // get_brand_wcpay_request:ok  发送成功
        WeixinJSBridge.log(res.err_msg);
        alert(res.err_code+res.err_desc);
    }
);
```

2.订单详情（package）

在商户调用JS API时，此时商户需要确定该笔订单详情，并将该订单详情通过一定的方式进行组合放入package。JS API调用后，微信将通过package的内容生成预支付单。订单详情由WxPayHelper类的get_cft_package（）方法生成。

package的所需字段列表以及签名方法如表5-7所示。需要注意：所有传入参数都是字符串类型！

表5-7　package订单详情字段

参数	名称	必填	格式	说明
bank_type	银行通道类型	是	字符串类型	固定为 "WX"，注意大写
body	商品描述	是	字符串类型，128 字节以下	商品描述
attach	附加数据	否	字符串类型，128 字节以下	附加数据，原样返回
partner	商户号	是	字符串类型	注册时分配的财付通商户号 partnerId
out_trade_no	商户订单号	是	字符串类型，32 字节以下	商户系统内部的订单号，32 个字符内，可包含字母，确保在商户系统唯一
total_fee	订单总金额	是	字符串类型	订单总金额，单位为分
fee_type	支付币种	是	字符串类型	取值：1（人民币），默认值是 "1"，暂只支持 1
notify_url	通知 URL	是	字符串类型，255 字节以下	在支付完成后，接收微信通知支付结果的 URL，需为绝对路径，255 字符内，格式如 http://wap.tenpay.com/tenpay.asp
spbill_create_ip	订单生成的机器 IP	是	字符串类型，15 字节以下	指用户浏览器端 IP，丌是商户服务器 IP，格式为 IPV4
time_start	交易起始时间	否	字符串类型，14 字节以下	订单生成时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010，时区为 GMT+8 beijing 该时间取自商户服务器
time_expire	交易结束时间	否	字符串类型，14 字节以下	订单失效时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 27 日 9 点 10 分 10 秒表示为 20091227091010，时区为 GMT+8 beijing 该时间取自商户服务器
transport_fee	物流费用	否	字符串类型	物流费用，单位为分。如果有值，必须保证 transport_fee+product_fee=total_fee
product_fee	商品费用	否	字符串类型	商品费用，单位为分。如果有值，必须保证 transport_fee+product_fee=total_fee
goods_tag	商品标记	否	字符串类型	商品标记，优惠券时可能用到
input_charset	传入参数字符编码	是	字符串类型	取值范围："GBK"、"UTF-8"，默认："GBK"

由于package中携带了生成订单的详细信息，因此微信将对package里面的内容进行鉴权，确定package携带的信息是真实、有效、合理的。因此，这里将定义生成package字符串的方法。package生成方法步骤如下所示。

假设以下为package传入参数：

```
bank_type=WX,
fee_type=1,
body=XXX,
input_charset=GBK,
partner=1900000109,
total_fee=1,
spbill_create_ip=127.0.0.1,
out_trade_no=16642817866003386000,
notify_url=http://www.qq.com
```

1）对所有传入参数按照字段名的ASCII码从小到大排序（字典序）后，使用URL键值对的格式（即key1=value1&key2=value2...）拼接成字符串string1，拼接后如下所示：

```
bank_type=WX&body=XXX&fee_type=1&input_charset=GBK&notify_url=http://www.qq.com&out_trade_no=16642817866003386000&partner=1900000109&spbill_create_ip=127.0.0.1&total_fee=1
```

2）在string1最后拼接上key=paternerKey得到stringSignTemp字符串，对stringSignTemp进行md5运算，再将得到的字符串所有字符转换为大写形式，得到sign值signValue，计算过程如下所示：

```
sign
=md5(string1&key=8934e7d15453e97507ef794cf7b0519d).toUpperCase
=md5(bank_type=WX&body=XXX&fee_type=1&input_charset=GBK&notify_url=http://www.qq.com&out_trade_no=16642817866003386000&partner=1900000109&spbill_create_ip=127.0.0.1&total_fee=1
="beef37ad19575d92e191c1e4b1474ca9".toUpperCase()
="BEEF37AD19575D92E191C1E4B1474CA9"
```

3）对string1中的所有键值对中的value迚行urlencode转码，按照1）步骤重新拼接成字符串，得到string2。对于JS前端程序，一定要使用函数encodeURIComponent进行urlencode编码（注意！进行urlencode时要将空格转化为“%20”而不是“+”）。

```
bank_type=WX&body=XXX&fee_type=1&input_charset=GBK&notify_url=http%3a%2f%2fwww.qq.com&out_trade_no=16642817866003386000&partner=1900000109&spbill_create_ip=127.0.0.1&total_fee=
```

4）将sign=signValue拼接到string1后面得到最终的package字符串。

```
bank_type=WX&body=XXX&fee_type=1&input_charset=GBK&notify_url=http%3a%2f%2fwww.qq.com&out_trade_no=16642817866003386000&partner=1900000109&spbill_create_ip=127.0.0.1&total_fee=
```

3.支付签名（paySign）

paySign字段是对本次发起JS API的行为进行鉴权，只有通过了paySign鉴权，才能继续对package鉴权并生成预支付单。这里将定义paySign的生成规则。参与paySign签名的字段包括：appid、timestamp、noncestr、package以及appkey（即paySignkey）。注意signType并不参与签名。paySign由WxPayHelper类的get_biz_sign（）方法生成。

生成paySign的具体过程如下。

假设参数如下：

```
"appId":"wxf8b4f85f3a794e77",
"timeStamp":"189026618",
"nonceStr":"adssdasssd13d",
"package":"bank_type=WX&body=XXX&fee_type=1&input_charset=GBK&notify_url=http%3a%2f%2fwww.qq.com&out_trade_no=16642817866003386000&partner=1900000109&spbill_create_ip=127.0.0.1
"appKey":"2Wozy2aksie1puXUBpWD8oZxiD1DfQuEaiC7KcRATv1Ino3mdopKaPGQQ7TtkNySuAmCaDCrw4xhPY5qKTBl7Fzm0RgR3c0WaVYIXZARsxzHV2x7iwPPzOz94dnwPWSn"
```

1）对所有待签名参数按照字段名的ASCII码从小到大排序（字典序）后，使用URL键值对的格式（即key1=value1&key2=value2…）拼接成字符串string1。需要注意的是所有参数名均为小写字符，例如appId在排序后字符串为appid。

```
appid=wxf8b4f85f3a794e77&appkey=2Wozy2aksie1puXUBpWD8oZxiD1DfQuEaiC7KcRATv1Ino3mdopKaPGQQ7TtkNySuAmCaDCrw4xhPY5qKTBl7Fzm0RgR3c0WaVYIXZARsxzHV2x7iwPPzOz94dnwPWSn&noncestr=adssda
```

2）对string1做签名算法，字段名和字段值都采用原始值，不进行URL转义。具体签名算法为paySign=SHA1（string）。签名后可得到：

```
paySign
=SHA1(appid=wxf8b4f85f3a794e77&appkey=2Wozy2aksie1puXUBpWD8oZxiD1DfQuEaiC7KcRATv1Ino3mdopKaPGQQ7TtkNySuAmCaDCrw4xhPY5qKTBl7Fzm0RgR3c0WaVYIXZARsxzHV2x7iwPPzOz94dnwPWSn&noncestr=
=7717231c335a05165b1874658306fa431fe9a0de
```

5.3.3　Native（原生）支付接口

Native（原生）支付过程中，首先需要商户定义符合Native（原生）支付规范的URL，也就是Native（原生）支付URL，同时在微信平台POST商户后台时需要提供package内容。

Native（原生）支付URL是一系列具有weixin：//wxpay/bizpayurl？前缀的URL，同时后面紧跟着一系列辨别商户的键值对。Native（原生）支付URL的规则如下：

```
weixin://wxpay/bizpayurl?sign=XXXXX&appid=XXXXXX&productid=XXXXXX&timestamp=XXXXXX&noncestr=XXXXXX
```

其中XXXXX为商户需要填写的内容，具体参数定义如表5-8所示。

表5-8　Native（原生）支付URL参数说明

参数	名称	必填	格式	说明
appid	公众账号 id	是	字符串类型	商户注册具有支付权限的公众账号成功后即可获得
timestamp	时间戳	是	字符串类型，32 字符以下	商户生成从 1970 年 1 月 1 日 00：00：00 至今的秒数，即当前的时间，且最终需要转换为字符串形式
noncestr	随机字符串	是	字符串类型，32 字符以下	商户生成的随机字符串
productid	商品唯一 id	是	字符串类型，32 字符以下	商户需要定义维护自己的商品 id，这个 id 与一张订单等价，微信平台凭借该 id 通过 POST 商户后台获取交易必需信息
sign	签名	是	字符串类型	对前面的其他字段与 appKey 按照字典序排序后，使用 SHA1 算法得到的结果。由商户生成后传入

1.Native（原生）支付URL签名

参与sign签名的字段包括：appid、timestamp、noncestr、productid以及appkey。

Native（原生）支付URL签名函数调用如下代码所示。

```
include_once("WxPayHelper.php");
$wxPayHelper = new WxPayHelper();
echo $wxPayHelper->create_native_url("1234");
```

该过程由wxPayHelper类中的create_native_url（）方法实现，详细实现过程如下所示。

假设参数如下：

```
"appid":"wxf8b4f85f3a794e77",
"timestamp":"189026618",
"noncestr":"adssdasssd13d",
"productid":"123456",
"appkey":"2Wozy2aksie1puXUBpWD8oZxiD1DfQuEaiC7KcRATv1Ino3mdopKaPGQQ7TtkNySuAmCaDCrw4xhPY5qKTBl7Fzm0RgR3c0WaVYIXZARsxzHV2x7iwPPzOz94dnwPWSn"
```

对所有待签名参数按照字段名的ASCII码从小到大排序（字典序）后，使用URL键值对的格式（即key1=value1&key2=value2…）拼接成字符串string1。这里需要注意的是所有参数名均为小写字符，即appId排序后的字符串则为appid，经过本过程得到：

```
string1="appid=wxf8b4f85f3a794e77&appkey=2Wozy2aksie1puXUBpWD8oZxiD1DfQuEaiC7KcRATv1Ino3mdopKaPGQQ7TtkNySuAmCaDCrw4xhPY5qKTBl7Fzm0RgR3c0WaVYIXZARsxzHV2x7iwPPzOz94dnwPWSn&nonces
```

对string1做签名算法，字段名和字段值都采用原始值，并进行URL转义。具体签名算法为sign=SHA1（string）。经过SHA1处理后得到：

```
sign= 18c6122878f0e946ae294e016eddda9468de80df
```

2.Native（原生）支付回调商户后台获取package

假设回调URL为http://www.outdomain.com/cgi-bin/bizpaygetpackage，在公众平台接到用户单击上述特殊Native（原生）支付的URL之后，会调用注册时填写的商户获取订单Package的回调URL。该Package由WxPayHelper类的create_native_package（）实现，完整调用代码如下：

```
include_once("WxPayHelper.php");
$commonUtil = new CommonUtil();
$wxPayHelper = new WxPayHelper();
$wxPayHelper->setParameter("bank_type", "WX");
$wxPayHelper->setParameter("body", "test");
$wxPayHelper->setParameter("partner", "1900000109");
$wxPayHelper->setParameter("out_trade_no", $commonUtil->create_noncestr());
$wxPayHelper->setParameter("total_fee", "1");
$wxPayHelper->setParameter("fee_type", "1");
$wxPayHelper->setParameter("notify_url", "htttp://www.baidu.com");
$wxPayHelper->setParameter("spbill_create_ip", "127.0.0.1");
$wxPayHelper->setParameter("input_charset", "GBK");
echo $wxPayHelper->create_native_package();
```

微信公众平台调用时会使用POST方式，推送XML格式的PostData，类似如下：

```
<xml>
    <OpenId><![CDATA[111222]]></OpenId>
    <AppId><![CDATA[wwwwb4f85f3a797777]]></AppId>
    <IsSubscribe>1</IsSubscribe>
    <ProductId><![CDATA[777111666]]></ProductId>
    <TimeStamp> 1369743908</TimeStamp>
    <NonceStr><![CDATA[YvMZOX28YQkoU1i4NdOnlXB1]]></NonceStr>
    <AppSignature><![CDATA[a9274e4032a0fec8285f147730d88400392acb9e]]></AppSignature>
    <SignMethod><![CDATA[sha1]]></SignMethod>
</xml>
```

PostData的参数说明如表5-9所示。

表5-9 PostData参数说明

参数	说明
OpenId	单击链接准备购买商品的用户 OpenId
AppId	公众账号的 appid
IsSubscribe	标记用户是否订阅该公众账号，1 为关注，0 为未关注
ProductId	第三方的商品 id 号
TimeStamp	时间戳
NonceStr	随机串
AppSignature	参数的加密签名，即 paySign 生成方法中所讲的签名方式生成的签名
SignMethod	签名方式，目前只支持“SHA1”，该字段不参与签名

第三方为了确保是来自于微信公众平台的合法请求，需要使用同样的方式生成签名，并与AppSignature的值进行对比。

参与签名的字段为：appid、appkey、productid、timestamp、noncestr、openid、issubscribe。为了返回Package数据，回调URL必须返回一个xml格式的返回数据，形如：

```
<xml>
    <AppId><![CDATA[wwwwb4f85f3a797777]]></AppId>
    <Package><![CDATA[a=1&url=http%3A%2F%2Fwww.qq.com]]></Package>
    <TimeStamp>1369745073</TimeStamp>
    <NonceStr><![CDATA[iuytxA0cH6PyTAVISB28]]></NonceStr>
    <RetCode>0</RetCode>
    <RetErrMsg><![CDATA[ok]]></RetErrMsg>
    <AppSignature><![CDATA[53cca9d47b883bd4a5c85a9300df3da0cb48565c]]></AppSignature>
    <SignMethod><![CDATA[sha1]]></SignMethod>
</xml>
```

其中，AppSignature依然是根据前文paySign所讲的签名方式生成的签名，参与签名的字段为：appid、appkey、package、timestamp、noncestr、retcode、reterrmsg。package的生成规则和JS API所定义的package生成规则一致。其中，对于一些第三方觉得商品已经过期或者其他错误的情况，可以在RetCode和RetErrMsg中体现出来，RetCode为0表示正确，可以定义其他错误；当定义其他错误时，可以在RetErrMsg中填上UTF8编码的错误提示信息，比如“该商品已经下架”，客户端会直接提示出来。

5.3.4 通知接口

1.交易通知

用户在成功完成支付后，微信平台通知（POST）商户服务器（notify_url）支付结果。商户可以使用notify_url的通知结果进行个性化页面的展示。

对后台通知交互时，如果微信收到商户的应答不是success或超时，微信不为通知失败，微信会通过一定的策略（如30分钟共8次）定期重新发起通知，尽可能提高通知的成功率，但微信不保证通知最终能成功。

由于存在重新发送后台通知的情况，因此同样的通知可能会多次发送给商户系统。商户系统必须能够正确处理重复的通知。

微信推荐的做法是，当收到通知进行处理时，首先检查对应业务数据的状态，判断该通知是否已经处理过，如果没有再进行处理，如果处理过直接返回success。在对业务数据进行状态检查和处理之前，要采用数据锁进行并发控制，以避免函数重入造成的数据混乱。

目前补单机制的间隔时间为：8s、10s、10s、30s、30s、60s、120s、360s、1000s。

后台通知通过请求中的notify_url进行，采用POST机制。返回通知中的参数一致，URL中字段说明如表5-10所示。

表5-10　通知接口参数说明

字段名	变量名	必填	类型	说明
协议参数				
签名方式	sign_type	否	String(8)	签名类型，取值：MD5、RSA，默认：MD5
接口版本	service_version	否	String(8)	版本号，默认为 1.0
字符集	input_charset	否	String(8)	字符编码，取值：GBK、UTF-8，默认：GBK
签名	sign	是	String(32)	签名
密钥序号	sign_key_index	否	Int	多密钥支持的密钥序号，默认 1
业务参数				
交易模式	trade_mode	是	Int	1- 即时到账，其他保留
交易状态	trade_state	是	Int	支付结果：0- 成功，其他保留
支付结果信息	pay_info	否	String(64)	支付结果信息，支付成功时为空
商户号	partner	是	String(10)	商户号，也即之前步骤的 partnerid，由微信统一分配的 10 位正整数 (120XXXXXXX) 号
付款银行	bank_type	是	String(16)	银行类型，在微信中使用 WX
银行订单号	bank_billno	否	String(32)	银行订单号
总金额	total_fee	是	Int	支付金额，单位为分，如果 discount 有值，通知的 total_fee+discount= 请求的 total_fee
币种	fee_type	是	Int	现金支付币种，目前只支持人民币，默认值是 1- 人民币
通知 ID	notify_id	是	String(128)	支付结果通知 id，对于某些特定商户，只返回通知 id，要求商户据此查询交易结果
订单号	transaction_id	是	String(28)	交易号，28 位长的数值，其中前 10 位为商户号，之后 8 位为订单产生的日期，如 20090415，最后 10 位是流水号
商户订单号	out_trade_no	是	String(32)	商户系统的订单号，不请求一致
商户数据包	attach	否	String(127)	商户数据包，原样返回
支付完成时间	time_end	是	String(14)	支付完成时间，格式为 yyyyMMddhhmmss，如 2009 年 12 月 27 日 9 点 10 分 10 秒表示为 20091227091010。时区为 GMT+8beijing
物流费用	transport_fee	否	Int	物流费用，单位分，默认 0。如果有值，必须保证 transport_fee+product_fee=total_fee
物品费用	product_fee	否	Int	物品费用，单位分。如果有值，必须保证 transport_fee+product_fee=total_fee
折扣价格	discount	否	Int	折扣价格，单位分，如果有值，通知的 total_fee+discount= 请求的 total_fee
买家别名	buyer_alias	否	String(64)	对应买家账号的一个加密串

同时，在postData中还将包含xml数据。数据如下：

```
<xml>
    <OpenId><![CDATA[111222]]></OpenId>
    <AppId><![CDATA[wwwwb4f85f3a797777]]></AppId>
    <IsSubscribe>1</IsSubscribe>
    <TimeStamp> 1369743511</TimeStamp>
```

```
    <NonceStr><![CDATA[jALldRTHAFd5Tgs5]]></NonceStr>
    <AppSignature><![CDATA[bafe07f060f22dcda0bfdb4b5ff756f973aecffa]]>
    </AppSignature>
    <SignMethod><![CDATA[sha1]]></SignMethod>
</xml>
```

各字段定义如表5-11所示。

表5-11　后台通知参数说明

参数	必填	说明
AppId	是	字段名称：公众账号 id；字段来源：商户注册具有支付权限的公众账号成功后即可获得；传入方式：由商户直接传入
TimeStamp	是	字段名称：时间戳；字段来源：商户生成从 1970 年 1 月 1 日 00:00:00 至今的秒数，即当前的时间；由商户生成后传入。取值范围：32 字符以下
NonceStr	是	字段名称：随机字符串；字段来源：商户生成的随机字符串；取值范围：长度为 32 个字符以下。由商户生成后传入。取值范围：32 字符以下
OpenId	是	支付该笔订单的用户 ID，商户可通过公众账号其他接口为付款用户服务
AppSignature	是	字段名称：签名；字段来源：对前面的其他字段不 appKey 按照字典序排序后，使用 SHA1 算法得到的结果。由商户生成后传入
IsSubscribe	是	用户是否关注了公众账号。1 为关注，0 为未关注

AppSignature依然是根据支付签名（paySign）签名方式生成，参与签名的字段为：appid、appkey、timestamp、noncestr、openid、issubscribe。

从以上信息可以看出，URL参数中携带订单相关信息，postData中携带该次支付的用户相关信息，这将便于商户拿到openid，以便后续提供更好的售后服务。

微信平台通过notify_url通知商户，商户做业务处理后，需要以字符串的形式反馈处理结果。结果为success时表示处理成功，微信系统收到此结果后不再进行后续通知；结果为fail或其他字符，微信收到此结果或者没有收到任何结果，系统通过补单机制再次通知。

下面定义了一段生成notifyargs字符串的示范过程。

假设以下为notifyargs传入参数：

```
bank_billno=206064184488,
bank_type=0,
discount=0,
fee_type=1,
input_charset=GBK,
notify_id=WE37gwCoFBcAKdkH34Y1nW94r_vao2ljmwE3oAHEeAP690xSVhRleOMfhsg
jwVGDpluT -vdS79kbDbkDnjYg4qsmTdSjuJxl,
out_trade_no=843254536943809900,
partner=1900000109,
product_fee=1,
sign_type=MD5,
time_end=20130606015331,
total_fee=1,
trade_mode=1,
trade_state=0,
transaction_id=1900000109201306060282555397,
transport_fee=0
```

1）除sign字段外，所有参数按照字段名的ASCII码从小到大排序后使用QueryString的格式（即key1=value1&key2=value2…）拼接而成字符串string1，空值不传递，不参与签名组串。经过URL键值对字典序排序后的字符串string1为：

```
bank_billno=206064184488&bank_type=0&discount=0&fee_type=1&input_charset=GBK&notify_id=WE37gwCoFBcAKdkH34Y1nW94r_vao2ljmwE3oAHEeAP690xSVhRleOMfhsgjwVGDpluT -vdS79kbDbkDnjYg4qsm
```

2）在string1最后拼接上key=paternerKey得到stringSignTemp字符串，并对stringSignTemp进行md5运算，再将得到的字符串所有字符转换为大写，得到sign值signValue。经过处理后得到sign为：

```
sign=md5(string1&key=8934e7d15453e97507ef794cf7b0519d).toUpperCase
=md5(bank_billno=206064184488&bank_type=0&discount=0&fee_type=1&input_charset=GBK&notify_id=WE37gwCoFBcAKdkH34Y1nW94r_vao2ljmwE3oAHEeAP690xSVhRleOMfhsgjwVGDpluT -vdS79kbDbkDnjY
=1900000109&product_fee=1&sign_type=MD5&time_end=20130606015331&total_fee=1&trade_mode=1&trade_state=0&transaction_id=1900000109201306060282555397&transport_fee=0&key=8934e7d15
="8ef1f69d5d9d4ec39d3787526f27924e".toUpperCase()
="8EF1F69D5D9D4EC39D3787526F27924E"
```

3）对string1进行urlencode转码，得到string2，经过处理，urlencode编码后得到：

```
bank_billno=206064184488&bank_type=0&discount=0&fee_type=1&input_charset=GBK&notify_id=WE37gwCoFBcAKdkH34Y1nW94r_vao2ljmwE3oAHEeAP690xSVhRleOMfhsgjwVGDpluT -vdS79kbDbkDnjYg4qsm
```

4）将sign=signValue拼接到string2后面得到最终的notifyargs字符串，结果：

```
bank_billno=206064184488&bank_type=0&discount=0&fee_type=1&input_charset=GBK&notify_id=WE37gwCoFBcAKdkH34Y1nW94r_vao2ljmwE3oAHEeAP690xSVhRleOMfhsgjwVGDpluT-vdS79kbDbkDnjYg4qsmT
```

2.发货通知delivernotify

为了更好地跟踪订单的情况，需要第三方在收到最终支付通知之后，调用发货通知API告知微信平台该订单的发货状态。

发货时间限制：虚拟、服务类24小时内，实物类72小时内。

请在收到支付通知后按时发货，并使用发货通知接口将相关信息同步到微信平台。若平台在规定时间内没有收到，将视作发货超时处理。

发货通知API的URL为：

```
https://api.weixin.qq.com/pay/delivernotify?access_token=xxxxxx
```

URL中的参数只包含目前微信公众平台凭证access_token，而发货通知的真正数据是放在PostData中的，格式如下：

```
{
    "appid" : "wwwwb4f85f3a797777",
    "openid" : "oX99MDgNcgwnz3zFN3DNmo8uwa-w",
    "transid" : "111112222233333",
    "out_trade_no" : "555666uuu",
    "deliver_timestamp" : "1369745073",
    "deliver_status" : "1",
    "deliver_msg" : "ok",
    "app_signature" : "53cca9d47b883bd4a5c85a9300df3da0cb48565c",
    "sign_method" : "sha1"
}
```

上述内容参数说明如表5-12所示。

表5-12　发货通知参数说明

参数	说明
appid	公众平台账户的 AppId
openid	购买用户的 OpenId，这个已经放在最终支付结果通知的 PostData 里了
transid	交易单号
out_trade_no	第三方订单号
deliver_timestamp	发货时间戳，这里指的是 Linux 时间戳
deliver_status	发货状态，1 表明成功，0 表明失败，失败时需要在 deliver_msg 填上失败原因
deliver_msg	发货状态信息，失败时可以填上 UTF8 编码的错误提示信息，比如“该商品已退款”
app_signature	根据支付签名（paySign）生成方法中所讲的签名方式生成的，参加签名字段为：appid、appkey、openid、transid、out_trade_no、deliver_timestamp、deliver_status、deliver_msg
sign_method	签名方法（不计入签名生成）

微信公众平台在校验ok之后，会返回数据表明是否通知成功，例如：{“errcode”：0，” errmsg”：” ok”}如果有异常，会在errcode和errmsg描述出来，如果成功errcode就为0。

3.订单查询orderquery

因为某一方技术的原因，可能导致商户在预期时间内都收不到最终支付通知，此时商户可以通过该API来查询订单的详细支付状态。

订单查询API的URL为：

```
https://api.weixin.qq.com/pay/orderquery?access_token=xxxxxx
```

URL中的参数只包含目前微信公众平台凭证access_token，而订单查询的真正数据是放在PostData中的，格式如下：

```
{
    "appid" : "wwwwb4f85f3a797777",
    "package" : "out_trade_no=11122&partner=1900090055&sign=4e8d0df3da0c3d0df38f",
    "timestamp" : "1369745073",
    "app_signature" : "53cca9d47b883bd4a5c85a9300df3da0cb48565c",
    "sign_method" : "sha1"
}
```

上述内容参数说明如表5-13所示。

表5-13　订单查询参数说明

参数	说明
appid	公众平台账户的 AppId
package	查询订单的关键信息数据，包含第三方唯一订单号 out_trade_no、财付通商户身份标识 partner（即前文所述的 partnerid）、签名 sign，其中 sign 是对参数字典序排序并使用 & 联合起来，最后加上 &key=partnerkey（唯一分配），进行 md5 运算，再转成全大写，最终得到 sign
timestamp	linux 时间戳
app_signature	根据支付签名（paySign）生成方法中所讲的签名方式生成的，参加签名字段为：appid、appkey、package、timestamp
sign_method	签名方法（不计入签名生成）

微信公众平台在校验ok之后，会返回数据表明是否通知成功，例如：

```
{"errcode":0,"errmsg":"ok",http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/..http://www.hzcourse.com/resource/readBook?p
```

如果有异常，会在errcode和errmsg描述出来，如果成功errcode就为0。

如果查询成功，会返回详细的订单数据，如下：

```
{
    "errcode":0,
    "errmsg":"ok",
    "order_info":
    {
        "ret_code":0,
        "ret_msg":"",
        "input_charset":"GBK",
        "trade_state":"0",
        "trade_mode":"1",
        "partner":"1900000109",
        "bank_type":"CMB_FP",
        "bank_billno":"207029722724",
        "total_fee":"1",
        "fee_type":"1",
        "transaction_id":"1900000109201307020305773741",
        "out_trade_no":"2986872580246457300",
        "is_split":"false",
        "is_refund":"false",
        "attach":"",
        "time_end":"20130702175943",
        "transport_fee":"0",
        "product_fee":"1",
        "discount":"0",
        "rmb_total_fee":""
    }
}
```

对于详细的订单信息，放在order_info中的json数据中，各个字段的含义如表5-14所示。

表5-14 订单数据参数说明

参数	说明
ret_code	查询结果状态码，0 表明成功，其他表明错误
ret_msg	查询结果出错信息
input_charset	返回信息中的编码方式
trade_state	订单状态，0 为成功，其他为失败
trade_mode	交易模式，1 为即时到账，其他保留
partner	财付通商户号，即前文的 partnerid
bank_type	银行类型
bank_billno	银行订单号
total_fee	总金额，单位为分
fee_type	币种，1 为人民币
transaction_id	财付通订单号
out_trade_no	第三方订单号
is_split	是否分账，false 为无分账，true 为有分账
is_refund	是否退款，false 为无退款，ture 为退款
attach	商户数据包，即生成订单 package 时商户填入的 attach
time_end	支付完成时间
transport_fee	物流费用，单位为分
product_fee	物品费用，单位为分
discount	折扣价格，单位为分
rmb_total_fee	换算成人民币之后的总金额，单位为分，一般看 total_fee 即可

4.告警通知

为了及时通知商户异常，提高商户在微信平台的服务质量。微信平台会向商户推送告警通知，包括发货延迟、调用失败、通知失败等情况，通知的地址是商户在申请支付时填写的告警通知URL，在“公众平台”→“服务”→“服务中心”→“商户功能”→“商户基本资料”→“告警通知URL”可以查看。商户接收到告警通知后需要尽快修复其中提到的问题，以免影响线上经营。

告警通知URL接收的postData中还将含xml数据，格式如下：

```
<xml>
    <AppId><![CDATA[wxf8b4f85f3a794e77]]></AppId>
    <ErrorType>1001</ErrorType>
    <Description><![CDATA[错识描述]]></Description>
    <AlarmContent><![CDATA[错误详情]]></AlarmContent>
    <TimeStamp>1393860740</TimeStamp>
    <AppSignature><![CDATA[f8164781a303f4d5a944a2dfc68411a8c7e4fbea]]></AppSignature>
    <SignMethod><![CDATA[sha1]]></SignMethod>
</xml>
```

5.3.5 收货地址共享接口

公众账号收货地址JS API接口为editAddress。

其参数说明如表5-15所示。

表5-15　收货地址接口参数说明

参数	必填	说明
appId	是	公众账号 AppID
scope	是	填写“jsapi_address”，获得编辑地址权限
signType	是	签名方式，目前仅支持 SHA1
addrSign	是	签名
timeStamp	是	时间戳
nonceStr	是	随机字符串

使用示例如下所示。

```
WeixinJSBridge.invoke('editAddress',{
        "appId" : getAppId(),
        "scope" : "jsapi_address",
        "signType" : "sha1",
        "addrSign" : "xxxxx",
        "timeStamp" : "12345",
        "nonceStr" : "10000",
    },function(res){
        //若res  中所带的返回值不为空，则表示用户选择该返回值作为收货地址。否则若返回空，则表示用户取消了这一次编辑收货地址。
        document.form1.address1.value =res.proviceFirstStageName;
        document.form1.address2.value =res.addressCitySecondStageName;
        document.form1.address3.value =res.addressCountiesThirdStageName;
        document.form1.detail.value = res.addressDetailInfo;
        document.form1.phone.value = res.telNumber;
    }
);
```

addrSign的生成规则如下所示。

假设参数如下：

```
appId=wx17ef1eaef46752cb
url=http://open.weixin.qq.com/
timeStamp=1384841012
nonceStr=123456
accessToken=OezXcEiiBSKSxW0eoylIeBFk1b8VbNtfWALJ5g6aMgZHaqZ
wK4euEskSn78Qd5pLsfQtuMdgmhajVM5QDm24W8X3tJ18kz5mhmkUcI
3RoLm7qGgh1cEnCHejWQo8s5L3VvsFAdawhFxUuLmgh5FRA
```

参与addrSign签名的字段包括：appId、url（当前网页URL）、timestamp、noncestr、accessToken（用户授权凭证）。这里scope、signType并不参与签名。

1）对所有待签名参数按照字段名的ASCII码从小到大排序（字典序）后，使用URL键值对的格式（即key1=value1&key2=value2…）拼接成字符串string1。这里需要注意的是所有参数名均为小写字符，例如appId在排序后字符串则为appid；经过键值对排序后得到string1为：

```
accesstoken=OezXcEiiBSKSxW0eoylIeBFk1b8VbNtfWALJ5g6aMgZHaqZwK4euEskSn78Qd5pLsfQtuMdgmhajVM5QDm24W8X3tJ18kz5mhmkUcI3RoLm7qGgh1cEnCHejWQo8s5L3VvsFAdawhFxUuLmgh5FRA&appId=wx17ef1e
```

2）对string1作签名算法，字段名和字段值都采用原始值，并进行URL转义。具体签名算法为addrSign=SHA1（string1）。经过签名后可得到：

```
addrSign=SHA1(accesstoken=OezXcEiiBSKSxW0eoylIeBFk1b8VbNtfWALJ5g6aMgZHaqZwK4euEskSn78Qd5pLsfQtuMdgmhajVM5QDm24W8X3tJ18kz5mhmkUcI3RoLm7qGgh1cEnCHejWQo8s5L3VvsFAdawhFxUuLmgh5FRA&
=ca604c740945587544a9cc25e58dd090f200e6fb
```

收货地址返回数据参数如表5-16所示。

表5-16　收货地址接口参数说明

返回值	说明
err_msg	edit_address:ok 获取编辑收货地址成功 edit_address:fail 获取编辑收货地址失败
userName	收货人姓名
telNumber	收货人电话
addressPostalCode	邮编
proviceFirstStageName	国标收货地址第一级地址
addressCitySecondStageName	国标收货地址第二级地址
addressCountiesThirdStageName	国标收货地址第三级地址
addressDetailInfo	详细收货地址信息
nationalCode	收货地址国家码

5.3.6　用户维权系统接口

用户在公众账号进行支付贩买行为出现异常时，通常会向腾讯客服投诉，因此微信需要即时了解公众账号与用户交易的详情。同时，为了最快效率地解决用户的问题，微信作为连接用户和商户的桥梁，会通过此客服系统即时将问题同步给商户，并将解决结果反馈至用户。

用户在新增投诉单后，微信平台通知（POST）商户服务器（payfeedback_url）支付结果。商户可以使用payfeedback_url的通知结果进行个性化页面的展示。注意，payfeedback_url请提交至微信相关接口人，微信登记后即可用。

后台通知通过请求中的payfeedback_url进行，采用POST机制。postData中将包含xml数据。数据如下所示。

1.用户新增投诉的xml

```
<xml>
    <OpenId><![CDATA[111222]]></OpenId>
    <AppId><![CDATA[wwwwb4f85f3a797777]]></AppId>
    <TimeStamp> 1369743511</TimeStamp>
    <MsgType><![CDATA[request]]></MsgType>
    <FeedBackId><![CDATA[5883726847655944563]]></FeedBackId>
    <TransId><![CDATA[10123312412321435345]]></TransId>
    <Reason><![CDATA[商品质量有问题]]></Reason>
    <Solution><![CDATA[补发货给我]]></Solution>
    <ExtInfo><![CDATA[明天六点前联系我 18610847266]]></ExtInfo>
    <AppSignature><![CDATA[bafe07f060f22dcda0bfdb4b5ff756f973aecffa]]></AppSignature>
    <SignMethod><![CDATA[sha1]]></SignMethod>
</xml>
```

2.用户确认处理完毕投诉的xml

```
<xml>
    <OpenId><![CDATA[111222]]></OpenId>
    <AppId><![CDATA[wwwwb4f85f3a797777]]></AppId>
    <TimeStamp>1369743511</TimeStamp>
    <MsgType><![CDATA[confirm/reject]]></MsgType>
    <FeedBackId><![CDATA[5883726847655944563]]></FeedBackId>
    <Reason><![CDATA[商品质量有问题]]></Reason>
    <AppSignature><![CDATA[bafe07f060f22dcda0bfdb4b5ff756f973aecffa]]></AppSignature>
    <SignMethod><![CDATA[sha1]]></SignMethod>
</xml>
```

上述数据各字段如表5-17所示。

表5-17　投诉通知数据参数说明

参数	必填	说明
AppId	是	字段名称：公众账号 id；字段来源：商户注册具有支付权限的公众账号成功后即可获得；传入方式：由商户直接传入
TimeStamp	是	字段名称：时间戳；字段来源：商户生成从 1970 年 1 月 1 日 00：00：00 至今的秒数，即当前的时间；由商户生成后传入。取值范围：32 字符以下
OpenId	是	支付该笔订单的用户 ID，商户可通过公众账号其他接口为付款用户服务
AppSignature	是	字段名称：签名；字段来源：对前面的其他字段不 appKey 按照字典序排序后，使用 SHA1 算法得到的结果。由商户生成后传入
MsgType	是	通知类型 request 用户提交投诉 confirm 用户确认消除投诉 reject 用户拒绝消除投诉

商户可以通过API调用，标记客户的投诉处理状态。

标记API的URL为：

```
https://api.weixin.qq.com/payfeedback/update?access_token=xxxxx&openid=XXXX&feedbackid=xxxx
```

URL中的参数包含目前微信公众平台凭证access_token，和客户投诉对应的单号feedbackid，以及openid。

微信公众平台在校验ok之后，会返回数据表明是否通知成功，例如：

```
{"errcode":0,"errmsg":"ok"}
```

如果有异常，会在errcode和errmsg描述出来，如果成功errcode就为0。

5.4　本章小结

本章介绍了微信支付的申请方法及功能介绍，最后介绍了微信支付接口的相关内容。

第6章　微信小店

2014年5月29日，微信公众平台宣布正式推出“微信小店”。微信小店是基于微信公众平台打造的原生电商模式，包括添加商品、商品管理、订单管理、货架管理、维权等功能，开发者可使用接口批量添加商品，快速开店。“微信小店”的上线，意味着微信公众平台上真正实现了技术“零门槛”的电商接入模式。

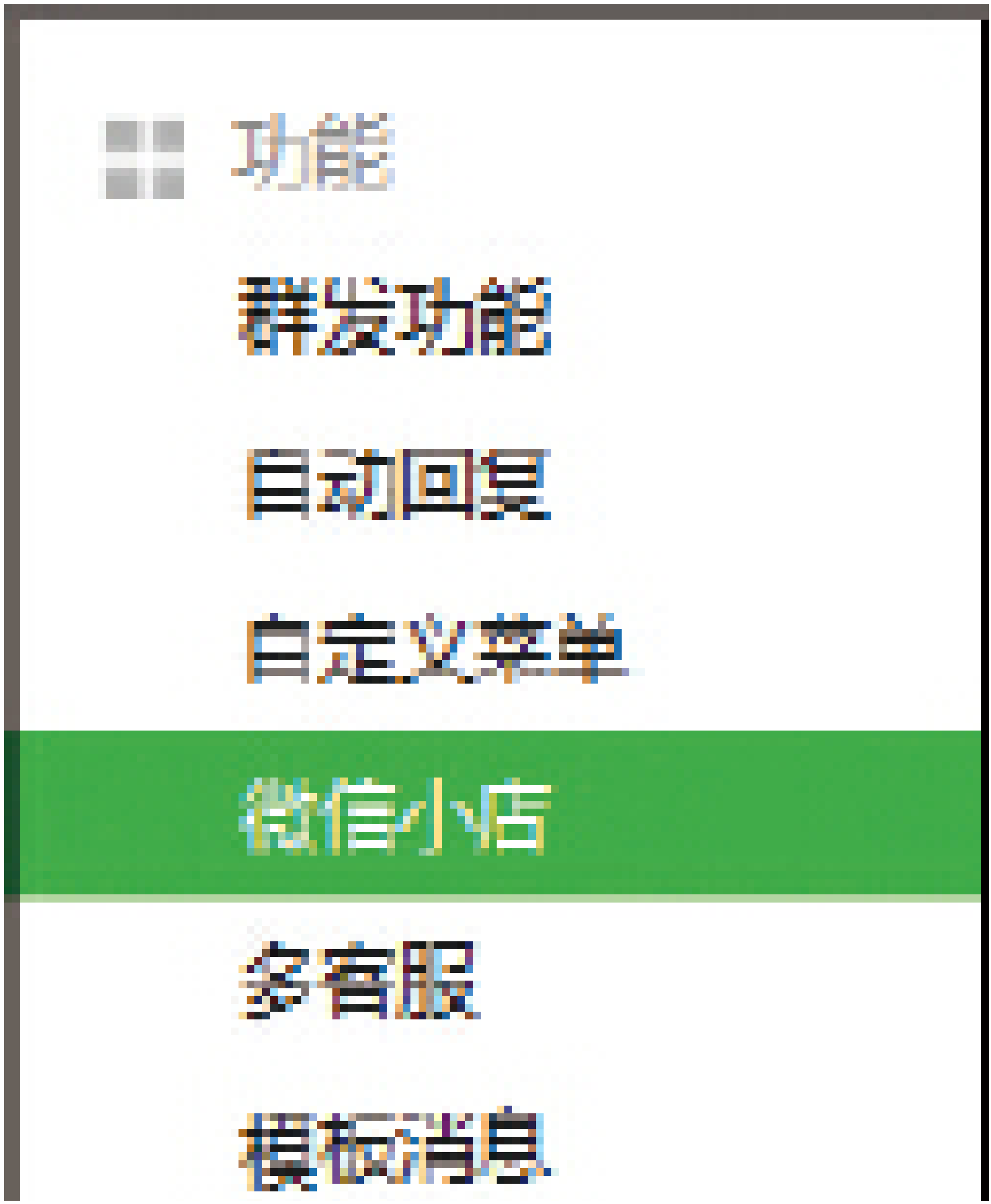
本章介绍微信小店搭建及常用功能的二次开发。

6.1 微信小店搭建

微信小店在微信支付能力的基础上，支持商家进行添加商品、商品管理、订单管理、货架管理、维权仲裁等操作。申请微信小店之前需要先申请微信支付，通过微信支付审核的商家，可以直接申请微信小店，通过审核后，就可以使用和运营微信小店了。

6.1.1 微信小店概况

微信小店可以从功能列表中的“添加功能插件”中进行添加，添加成功后，将在功能列表中显示，如图6-1所示。



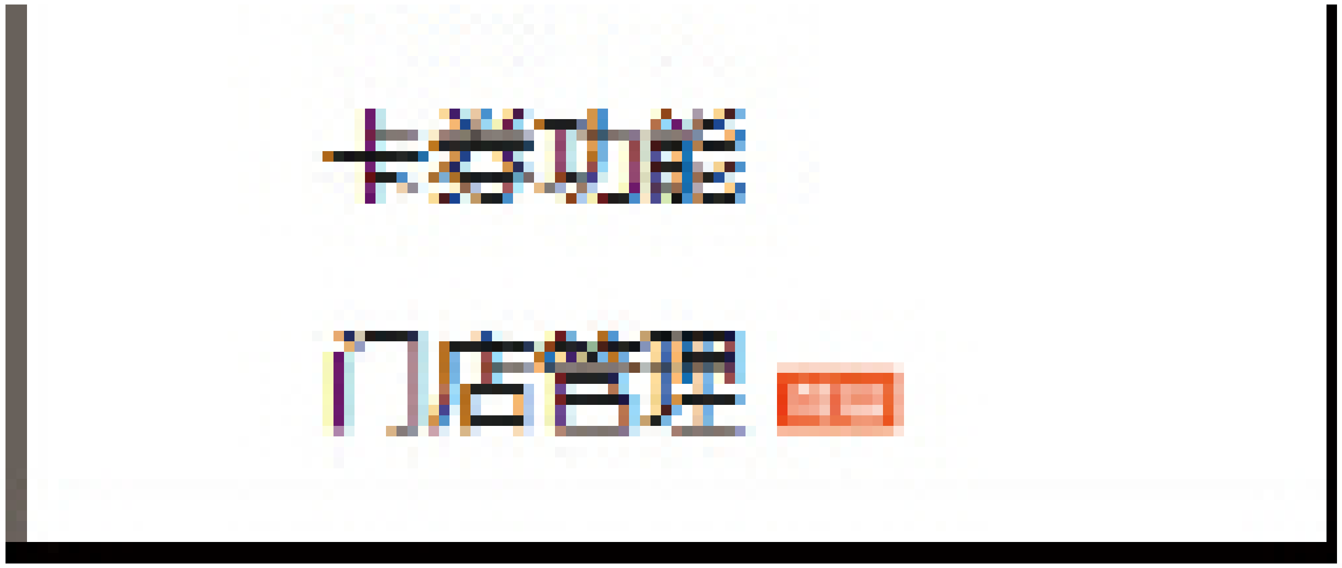


图6-1　微信功能列表

单击“微信小店”标签将进入微信小店后台管理界面，可以看到微信小店的概况，例如待发货订单及待处理维权/仲裁单数、昨日关键指标等，如图6-2所示。

图6-2　微信小店概况

除此之外，还有关键指标趋势图及关键指标明细等图表，如图6-3和图6-4所示。

图6-3　关键指标趋势图

关键指标明细 下载　最近30日　最近15日　最近7日　2015-01-12 至 2015-01-18

日期	订单数	成交商品数	成交额	商品浏览量	货架浏览量	小店访问人数
2015-01-18	0	0	0	0	1	1
2015-01-17	0	0	0	0	0	0
2015-01-16	0	0	0	4	9	6
2015-01-15	0	0	0	0	3	3
2015-01-14	1	1	98	5	9	3
2015-01-13	0	0	0	0	0	0
2015-01-12	0	0	0	0	2	2

图6-4　关键指标明细

6.1.2　运费模板管理

运营微信小店之前，需要对它进行一些配置，首先可以配置运费模板。

运费模板就是为一批商品设置同一个运费。当您需要修改运费的时候，这些关联商品的运费将一起被修改。单击标签列表中的“运费模板管理”可以新建及编辑运费模板，如图6-5所示。

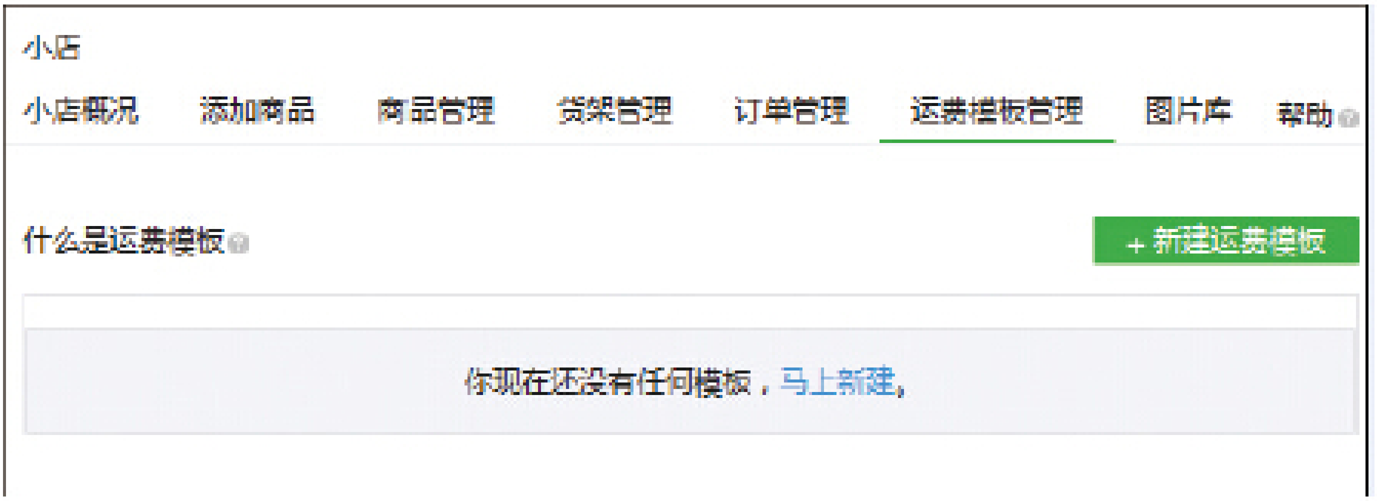


图6-5　运费模板管理

单击“新建运费模板”将进入到新建窗口中，可在窗口中填写模板名称，选择配送方式及运费设置，如图6-6所示。

图6-6　新建运费模板

填完后，单击“保存”按钮，将看到当前已建好的运费模板，如图6-7所示。

图6-7　已建好的运费模板

6.1.3　商品分组管理

建好运费模板之后，可以创建商品的分类。合理设置商品分类能方便将商品填写到货架，并及时进行管理。

单击标签列表中“商品管理”，进入到“商品分组管理”界面中，在右侧的商品分组列表中，单击“新建分组”，如图6-8所示。

图6-8　新建分组

在新建分组窗口中填写分组名称，单击保存后，将成功建立一个商品分组，如图6-9所示。

图6-9　分组名称

6.1.4　图片库

图片库用于存储商品的图片，一个商品至少需要上传一张主图，商品主图将会作为商品的默认图片出现在货架及商品详情页，还可以上传多张其他图片，方便用户更好地了解您的商品。微信小店的图片库存储在微信素材中。

在微信小店功能标签中，单击图片库，如图6-10所示。

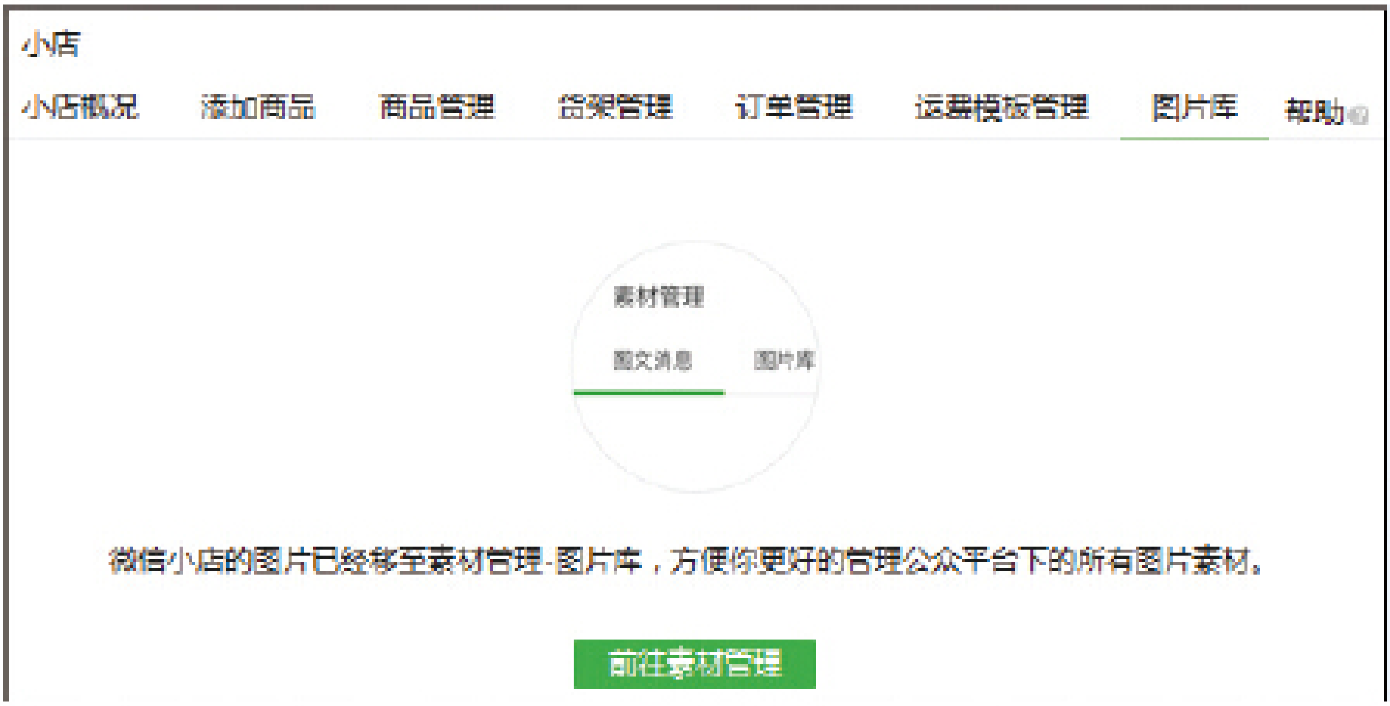


图6-10　图片库

系统将提示前面素材管理进行上传，如图6-11所示。

图6-11　素材管理中的图片库

单击“上传”按钮，将弹出图片选择窗口，如图6-12所示。

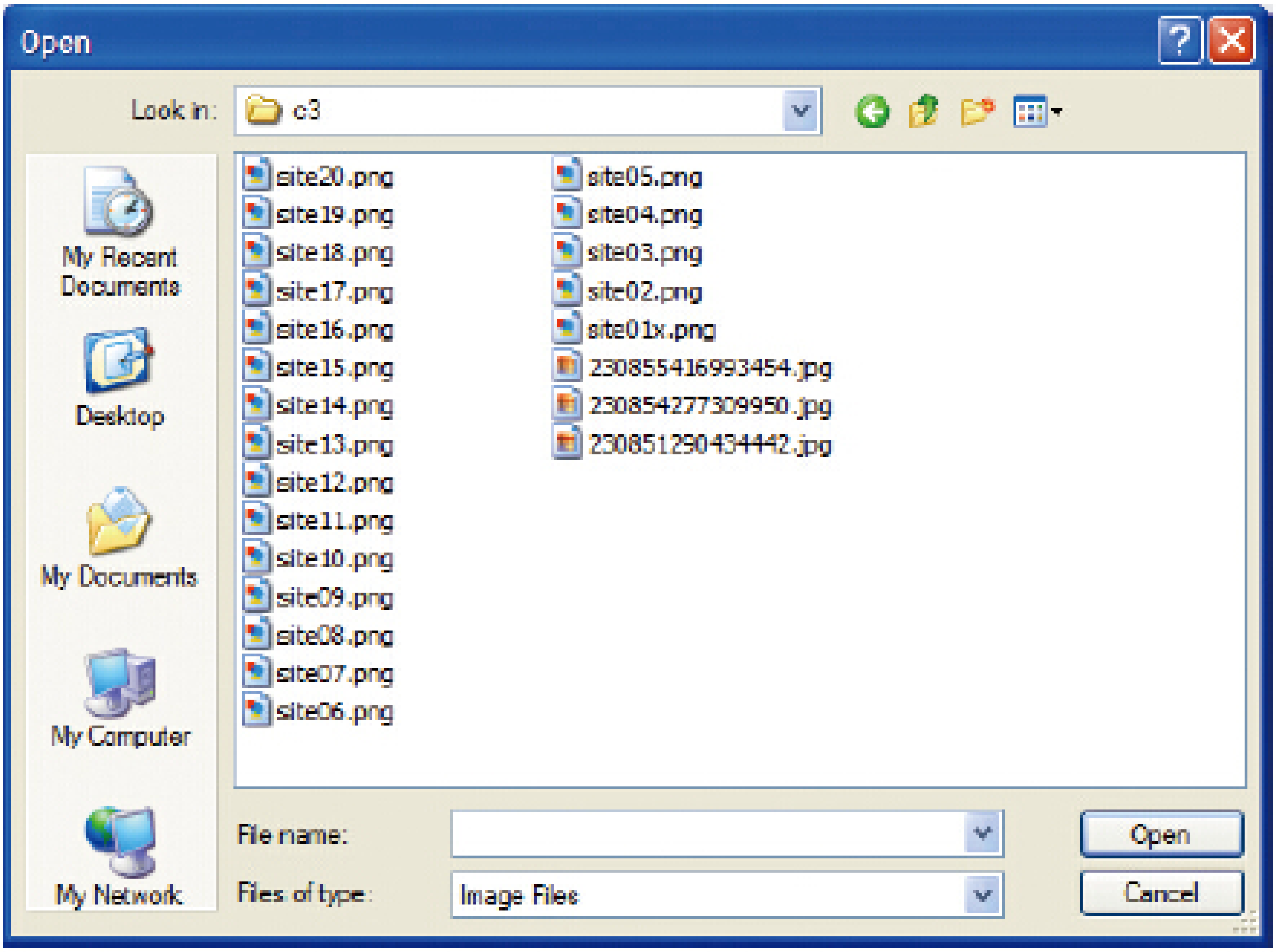


图6-12　图片选择窗口

选择要上传的图片，上传后将显示在图片库中，如图6-13所示。

图6-13　图片库

6.1.5　添加商品

在运费模板、商品分组、图片库都上传好了相应的内容后，就可以开始添加商品了。

单击功能列表中的“添加商品链接”，进入到“类目选择”界面，根据自己产品实际情况选择相应的类目，如图6-14所示。

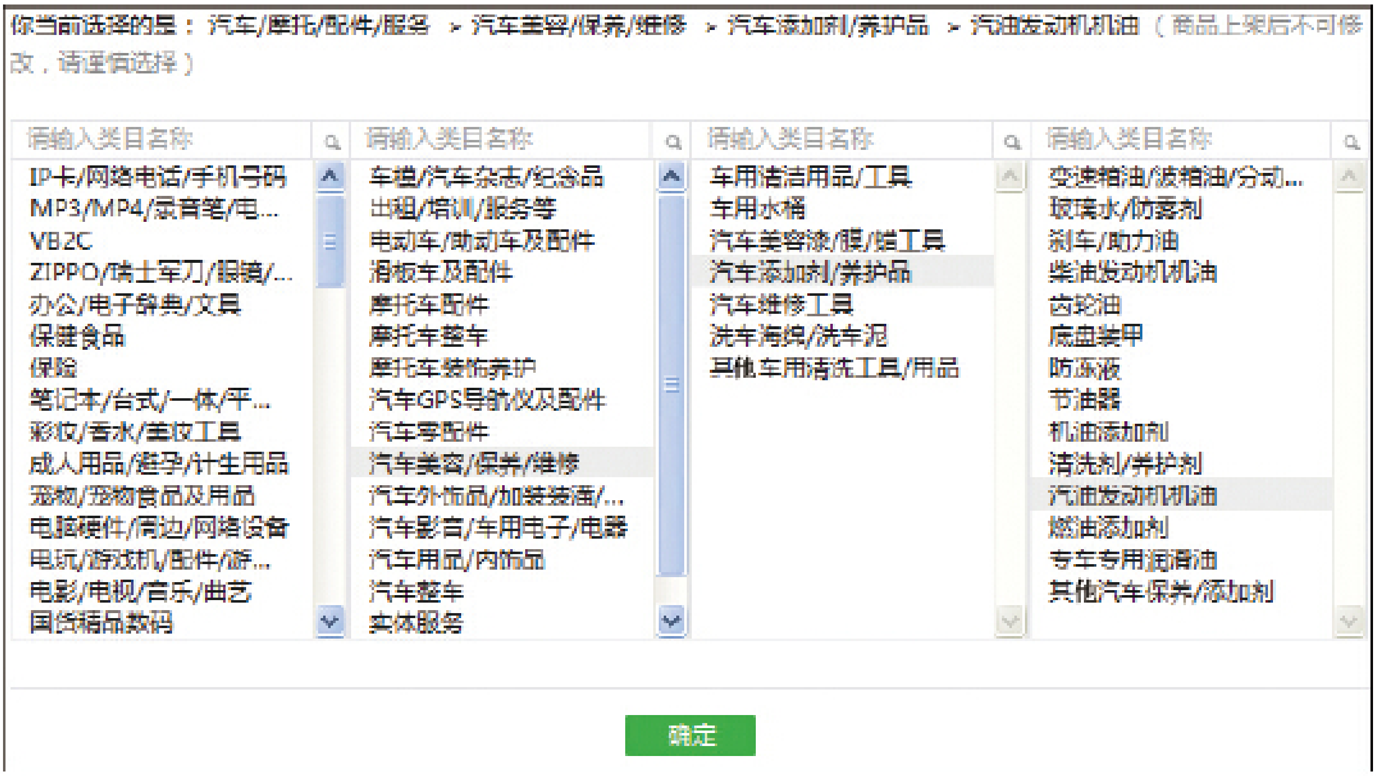


图6-14 类目选择

选择好类目之后，单击“确定”按钮，进入商品信息填写界面，可以选择商品属性及商品分组，如图6-15所示。

图6-15 商品属性及分组

接下来填写商品名称、微信价格、商品库存等信息，其他内容可选填或根据产品实际情况修改，如图6-16所示。

图6-16　商品基本信息

然后，为商品选择图片。单击主图或其他图片中的图片选择控件将弹出图片库，在图片库中单击选择图片后单击“确认”按钮即可，如图6-17所示。

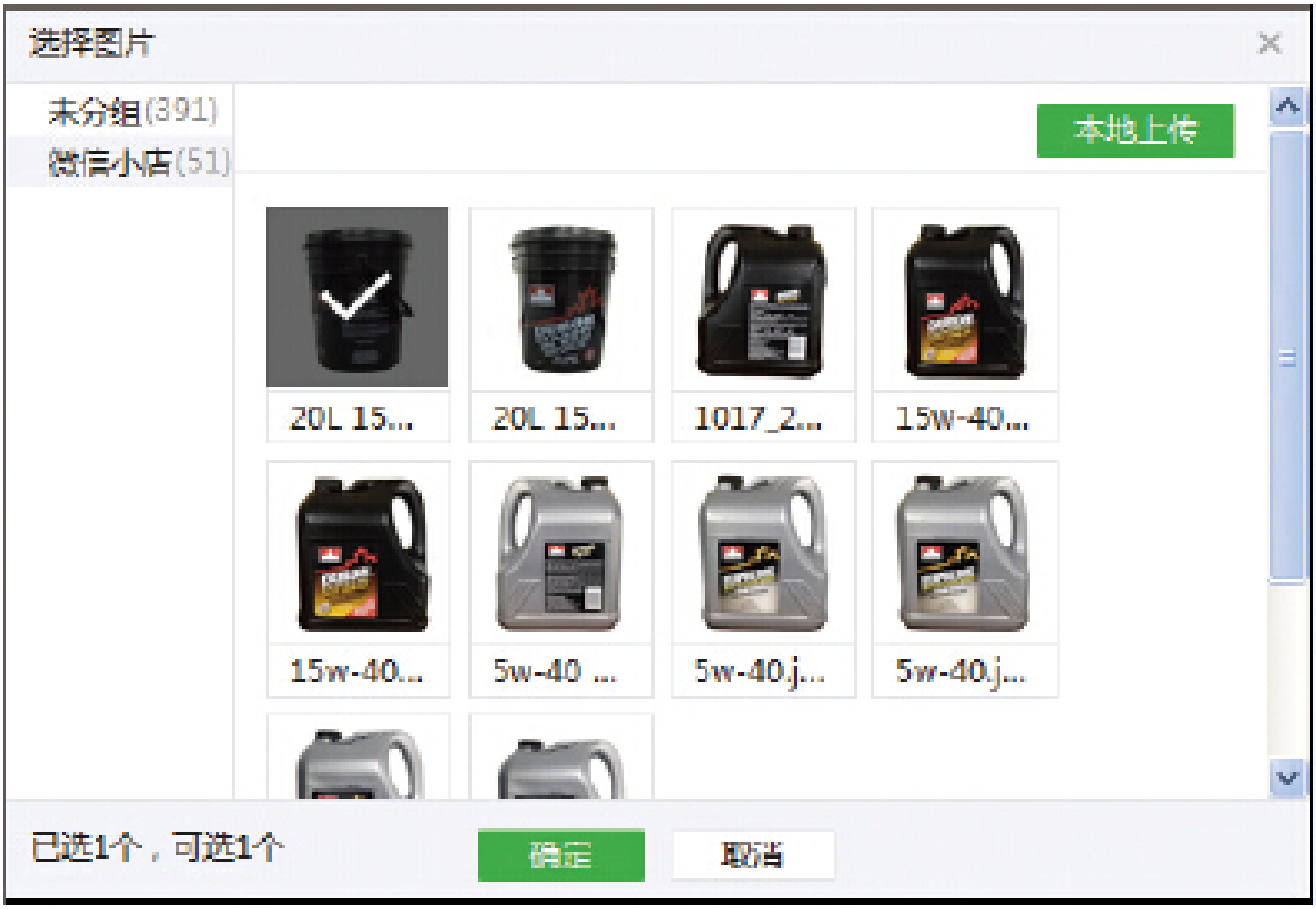


图6-17　选择图片

确认后将会显示当前已选择的图片，如图6-18所示。

图6-18　已选择的图片

最后，设置商品的物流信息、售后信息及上架设置等内容，如图6-19所示。

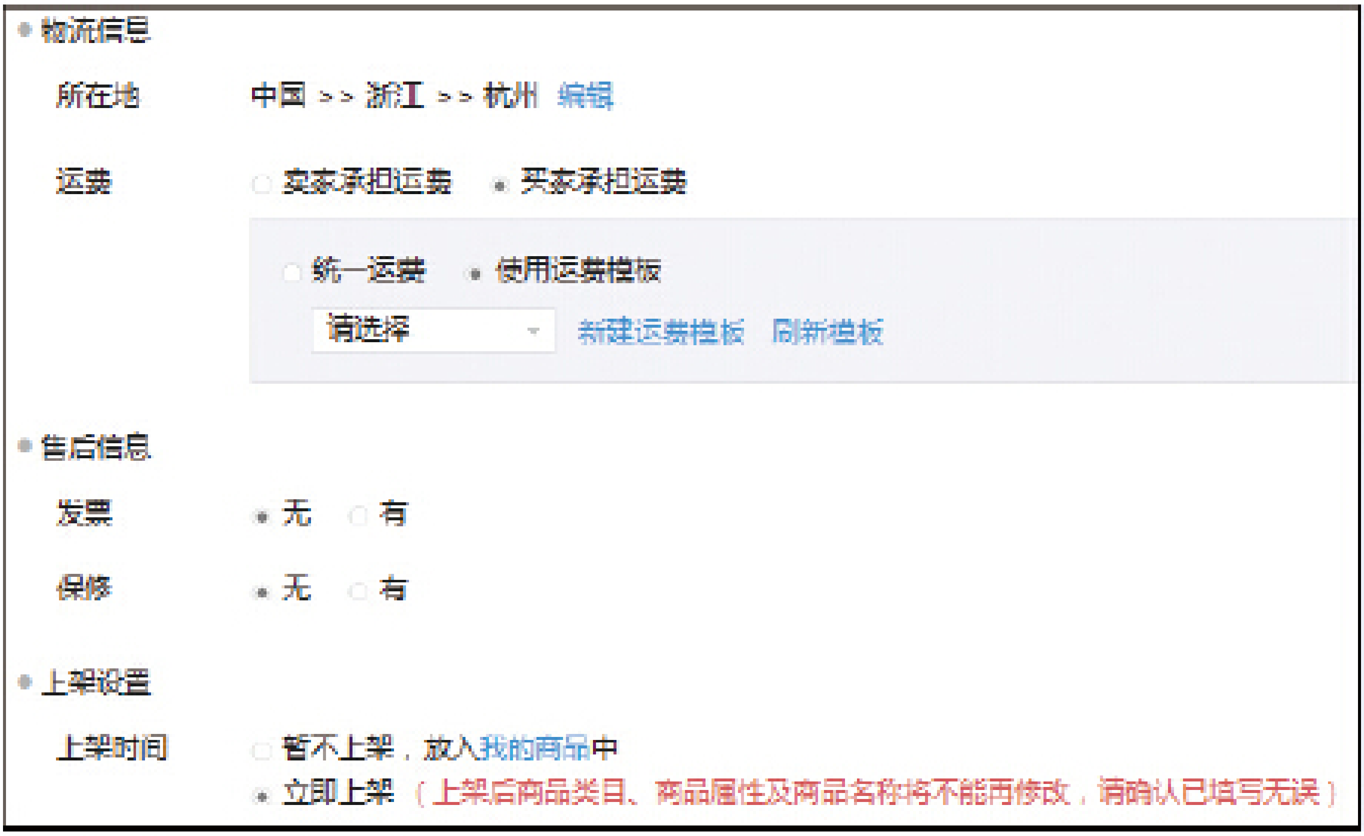


图6-19　物流信息/售后信息/上架设置

所有内容都设置好并确定提交之后，可以在商品管理中看到刚才已上传的商品，如图6-20所示。

图6-20　已上架商品

6.1.6　货架管理

货架是商家用于承载商品的模板，每一个货架是由不同的控件组成的。

单击功能列表中的“添加货架管理”链接后，将显示当前的货架情况，如图6-21所示。

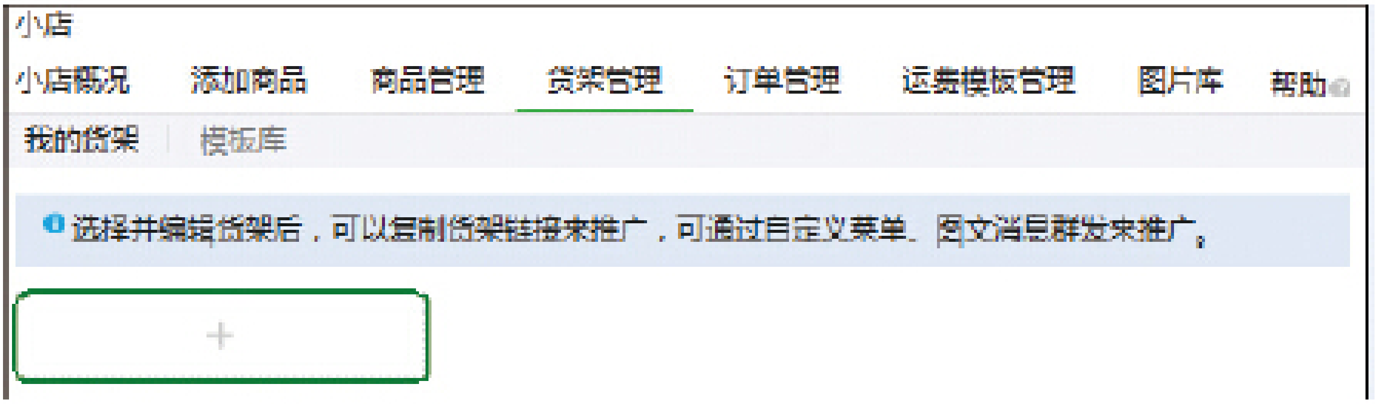


图6-21　我的货架

单击下方的添加控件“+”，将显示当前的货架模板，如图6-22所示。

图6-22　货架模板

选择其中一个使用，将进入该货架的编辑界面，如图6-23所示。

分组名称
商品名称
¥1 ¥2
商品名称
¥1 ¥2
商品名称
商品名称

图6-23　编辑货架

将鼠标移到招牌处，将出来一个笔形图标，单击该图片可以上传小店招牌，图片的建议尺寸为640像素×300像素，如图6-24所示。

图6-24　选择小店招牌

在下部分的商品展示组件中，为组件选择要展示的商品分组名称及展示商品数量，如图6-25所示。

图6-25　商品展示组件配置

上述横幅和展示组件均配置好并保存后，商品的货架如图6-26所示。

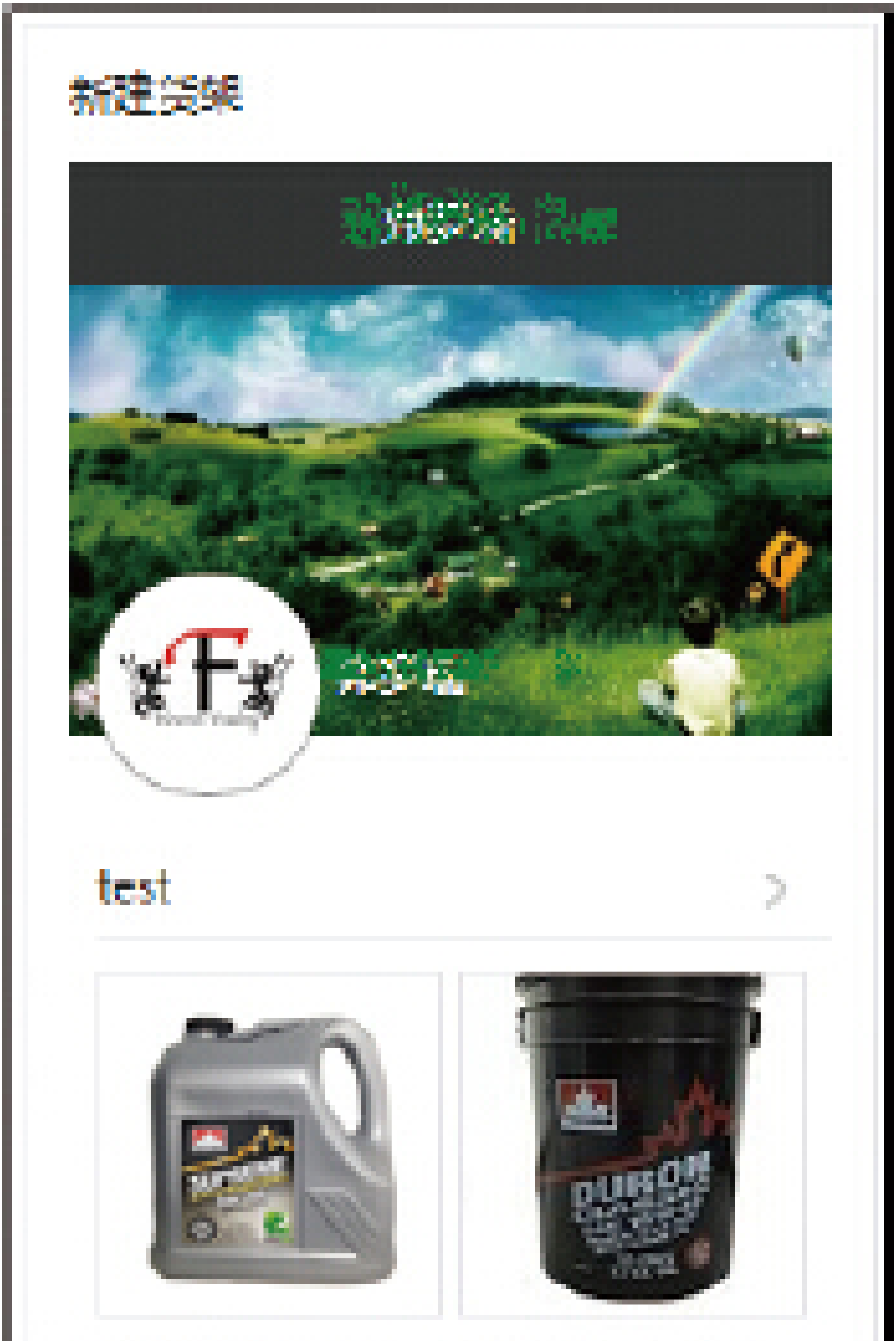


图6-26　商品货架

鼠标移到底部环形别针图标处，可以看到复制链接字样，单击别针图标，该链接将被复制到剪切板中。可以将链接放入自定义菜单中，这样用户就可以从菜单中进入微信小店购买商品。

6.2　微信小店二次开发

微信小店已经具备一个商城的基本功能，包括商品管理、订单管理、货架管理、运费管理等功能，但还不够完善，比如没有购物车、订单查询等功能。根据微信小店提供的API接口及第三方接口，可以开发其他几个常用的功能，包括付款交易通知、订单明细查询以及快递查询功能，另外还可以在菜单中加入维权功能。

6.2.1　微信小店SDK

在正式讲解开发内容之前，我们需要先将常用功能函数写入类中，以便在后面的章节中调用。以下是方倍工作室开发的用于微信小店的SDK代码。

```
<?php

/*
    方倍工作室
    CopyRight 2014 All Rights Reserved
*/
require_once('config.php');   //引用配置

class class_weixin
{
    var $appid = APPID;
    var $appsecret = APPSECRET;

    //构造函数，获取Access Token
    public function __construct($appid = NULL, $appsecret = NULL)
    {
        if($appid && $appsecret){
            $this->appid = $appid;
            $this->appsecret = $appsecret;
        }

        $url = "https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=".$this->appid."&secret=".$this->appsecret;
        $res = $this->http_request($url);
        $result = json_decode($res, true);
        //save to Database or Memcache
        $this->access_token = $result["access_token"];
        $this->lasttime = time();
    }

    //创建菜单
    public function create_menu($data)
    {
        $url = "https://api.weixin.qq.com/cgi-bin/menu/create?access_token=".
$this->access_token;
        $res = $this->http_request($url, $data);
        return json_decode($res, true);
    }

    //根据订单ID获取订单详情
    public function get_detail_by_order_id($id)
    {
        $data = array('order_id' =>$id);
        $url = "https://api.weixin.qq.com/merchant/order/getbyid?access_token=".
$this->access_token;
        $res = $this->http_request($url, json_encode($data));
        return json_decode($res, true);
    }

    //根据订单状态/创建时间获取订单详情
    public function get_detail_by_filter($data = null)
    {
        $url = "https://api.weixin.qq.com/merchant/order/getbyfilter?access_token=".
$this->access_token;
        $res = $this->http_request($url, $data);
        return json_decode($res, true);
    }

    //发送客服消息，已实现发送文本，其他类型可扩展
    public function send_custom_message($touser, $type, $data)
    {
        $msg = array('touser' =>$touser);
        $msg['msgtype'] = $type;
        switch($type)
        {
            case 'text':
                $msg[$type]    = array('content'=>urlencode($data));
                break;
```

```
            case 'news':
                $msg[$type]    = array('articles'=>$data);
                break;
            default:
                $msg['text']    = array('content'=>urlencode("不支持的消息类型 ".$type));
                break;
        }
        $url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$this->access_token;
        return $this->http_request($url, urldecode(json_encode($msg)));
    }

    //发送模板消息
    public function send_template_message($data)
    {
        $url = "https://api.weixin.qq.com/cgi-bin/message/template/send?access_token=".$this->access_token;
        $res = $this->http_request($url, $data);
        return json_decode($res, true);
    }

    //https请求（支持GET和POST）
    protected function http_request($url, $data = null)
    {
        $curl = curl_init();
        curl_setopt($curl, CURLOPT_URL, $url);
        curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
        curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
        if (!empty($data)){
            curl_setopt($curl, CURLOPT_POST, 1);
            curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
        }
        curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
        $output = curl_exec($curl);
        curl_close($curl);
        return $output;
    }
}
```

上述代码定义了微信小店的类，在类中定义了本章开发实现需要用到的方法，包括前面章节提到的创建自定义菜单、客服消息及模板消息等功能，以及需要用到的根据订单ID获取订单详情和根据订单状态/创建时间获取订单详情两个方法，这两个方法的使用方法在后面的章节将详细介绍。

6.2.2　交易付款通知

用户在微信中付款成功后，微信服务器会将订单付款通知推送到开发者在公众平台网站中设置的回调URL（在开发模式中设置）中。该通知是一个merchant_order事件通知。

微信推送的merchant_order事件消息内容如下：

```
<xml>
    <ToUserName><![CDATA[weixin_media1]]></ToUserName>
    <FromUserName><![CDATA[oDF3iYyVlek46AyTBbMRVV8VZVlI]]></FromUserName>
    <CreateTime>1398144192</CreateTime>
    <MsgType><![CDATA[event]]></MsgType>
    <Event><![CDATA[merchant_order]]></Event>
    <OrderId><![CDATA[7197417460812584720]]></OrderId>
    <OrderStatus>2</OrderStatus>
    <ProductId><![CDATA[pDF3iYx7KDQVGzB7kDg6Tge5OKFo]]></ProductId>
    <SkuInfo><![CDATA[10001:1000012;10002:100021]]></SkuInfo>
</xml>
```

其中的OrderId参数即为该次交易的订单ID号。我们可以根据订单ID查询订单详情。

根据订单ID查询订单详情接口如下所示：

```
https://api.weixin.qq.com/merchant/order/getbyid?access_token=ACCESS_TOKEN
```

该接口的POST数据格式如下，参数说明如表6-1所示。

```
{
    "order_id": "7197417460812584720"
}
```

表6-1　订单查询接口参数说明

字段	说明
order_id	订单 ID

返回数据格式如下：

```
{
    "errcode": 0,
    "errmsg": "success",
    "order": {
        "order_id": "7197417460812533543",
        "order_status": 6,
        "order_total_price": 69,
        "order_create_time": 1394635817,
        "order_express_price": 5,
        "buyer_openid": "oDF3iY17NsDAW4UP2qzJXPsz1S9Q",
        "buyer_nick": "方倍",
        "receiver_name": "方倍工作室",
        "receiver_province": "广东省",
        "receiver_city": "深圳市",
        "receiver_address": "华景路一号南方通信大厦5楼",
        "receiver_mobile": "123456789",
        "receiver_phone": "123456789",
        "product_id": "pDF3iYx7KDQVGzB7kDg6Tge5OKFo",
        "product_name": "包邮正版 《微信公众平台开发最佳实践》 双十一特惠",
        "product_price": 1,
        "product_sku": "10000983:10000995;10001007:10001010",
        "product_count": 1,
        "product_img": "http://img2.paipaiimg.com/00000000/item-52B87243-63CCF66C00000000040100003565C1EA.0.300x300.jpg",
        "delivery_id": "1900659372473",
        "delivery_company": "059Yunda",
        "trans_id": "1900000109201404103172199813"
    }
}
```

上述数据的参数说明如表6-2所示。

表6-2　订单查询接口返回内容参数说明

字段		说明
errcode		错误码
errmsg		错误信息
order		订单详情
	order_id	订单 ID
	order_status	订单状态
	order_total_price	订单总价格（单位：分）
	order_create_time	订单创建时间
	order_express_price	订单运费价格（单位：分）
	buyer_openid	买家微信 OPENID
	buyer_nick	买家微信昵称
	receiver_name	收货人姓名
	receiver_province	收货地址省份
	receiver_city	收货地址城市
	receiver_address	收货详细地址
	receiver_mobile	收货人移动电话
	receiver_phone	收货人固定电话
	product_id	商品 ID
	product_name	商品名称
	product_price	商品价格（单位：分）
	product_sku	商品 SKU
	product_count	商品个数
	product_img	商品图片
	delivery_id	运单 ID
	delivery_company	物流公司编码
	trans_id	交易 ID

付款通知在事件消息中的实现代码如下所示。

```
//接收事件消息
private function receiveEvent($object)
{
    $content = "";
    switch ($object->Event)
    {
        case "subscribe":
            $content = "欢迎关注方倍工作室";
            break;
        case "merchant_order":
            $orderid = strval($object->OrderId);
            $openid = strval($object->FromUserName);
            require_once('weixin.class.php');
            $weixin = new class_weixin();
            $orderArr0 = $weixin->get_detail_by_order_id($orderid);
            $orderArr  = $orderArr0["order"];

            //模板消息发送 消费品 - 消费品 - 购买成功通知
            $template = array('touser' => $openid,
            'template_id' => "-dEAy_IB2iUY8aePSMHfnxpJvxKf1rwuMd_4tLNyGF4",
            'url' => "",
            'topcolor' => "#7B68EE",
            'data' => array('first'=> array('value' => urlencode("您好，方倍，欢迎使用模板消息。"),
                                            'color' => "#000000",
                                            ),
                                'product'=> array('value' => urlencode($orderArr
["product_name"]),
                                            'color' => "#000093",
                                            ),
                                'price'  => array('value' => urlencode("￥".($orderArr
["order_total_price"] / 100)),
                                            'color' => "#FF0000",
                                            ),
                                'time'   => array('value' => urlencode(date("Y-m-d H:i:s",
($orderArr["order_create_time"]))),
                                            'color' => "#006000",
                                            ),
                                'remark' => array('value' => urlencode("你的订单已提交，我们将尽快发货，祝您生活愉快！"),
                                            'color' => "#000000",
                                            ),
                                )
            );
            $result = $weixin->send_template_message(urldecode(json_encode($template)));

            $content = "";
```

```
43             break;
44         default:
45             $content = "";
46             break;
47 
48     }
49     if(is_array($content)){
50         if (isset($content[0])){
51             $result = $this->transmitNews($object, $content);
52         }else if (isset($content['MusicUrl'])){
53             $result = $this->transmitMusic($object, $content);
54         }
55     }else{
56         $result = $this->transmitText($object, $content);
57     }
58 
59     return $result;
60 }
```

上述代码中和订单查询通知相关部分简要说明如下：

第10行：判断是否收到订单付款通知。

第11行~第12行：获取用户的OpenID及订单ID。

第13行~第16行：引用微信小店SDK，根据订单ID查询订单详情。

第18行~第40行：将订单详情中的内容填充到微信模板消息中并发送。

最后，一个购买成功通知的模板消息实现如图6-27所示。

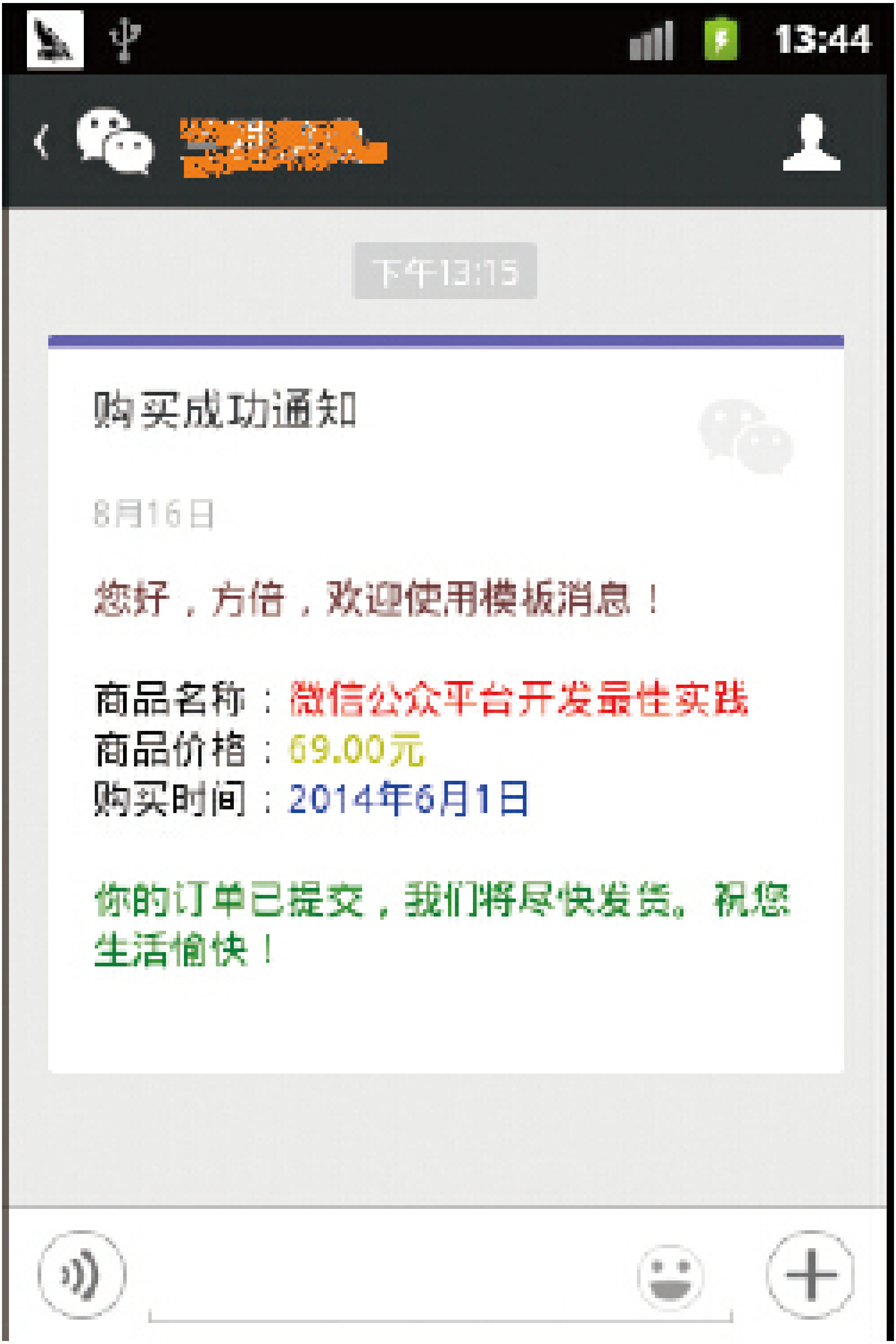

13:44
下午13:15
购买成功通知
8月16日
您好，方倍，欢迎使用模板消息！
商品名称：微信公众平台开发最佳实践
商品价格：69.00元
购买时间：2014年6月1日
你的订单已提交，我们将尽快发货。祝您生活愉快！

图6-27 购买成功通知

6.2.3 我的订单查询

微信小店的后台中提供订单查询功能，但那是给商家看的，如果用户需要查看自己的订单，则要使用微信小店的接口来开发实现。这需要用到根据订单状态/创建时间获取订单详情接口。

根据订单状态/创建时间获取订单详情接口如下所示：

```
https://api.weixin.qq.com/merchant/order/getbyid?access_token=ACCESS_TOKEN
```

该接口的POST数据格式如下，参数说明如表6-3所示。

```
{
    "status": 2,
    "begintime": 1397130460,
    "endtime": 1397130470
}
```

表6-3 订单查询接口参数说明

字段	说明
status	订单状态（不带该字段 - 全部状态，2- 待发货，3- 已发货，5- 已完成，8- 维权中）
begintime	订单创建时间起始时间（不带该字段则不按照时间做筛选）
endtime	订单创建时间终止时间（不带该字段则不按照时间做筛选）

返回数据格式如下，参数说明如表6-4所示。

```
{
    "errcode": 0,
    "errmsg": "success",
    "order_list": [
        {
          "order_id": "7197417460812533543",
          "order_status": 6,
          "order_total_price": 6,
          "order_create_time": 1394635817,
          "order_express_price": 5,
          "buyer_openid": "oDF3iY17NsDAW4UP2qzJXPsz1S9Q",
          "buyer_nick": "likeacat",
          "receiver_name": "方倍",
          "receiver_province": "广东省",
          "receiver_city": "广州市",
          "receiver_address": "华景路一号南方通信大厦5楼",
          "receiver_mobile": "123456",
          "receiver_phone": "123456",
          "product_id": "pDF3iYx7KDQVGzB7kDg6Tge5OKFo",
          "product_name": "《微信公众平台开发最佳实践》",
          "product_price": 1,
          "product_sku": "10000983:10000995;10001007:10001010",
          "product_count": 1,
          "product_img": "http://mmbiz.qpic.cn/mmbiz/4whpV1VZl2icND8WwMThBEcehjh
Dv2icY4GrDSG5RLM3B2qd9kOicWGVJcsAhvXfibhWRNoGOvCfMC33G9z5yQr2Qw/0",
          "delivery_id": "1900659372473",
          "delivery_company": "059Yunda",
          "trans_id": "1900000109201404103172199813"
        },
        {
          "order_id": "7197417460812533569",
          "order_status": 8,
          "order_total_price": 1,
          "order_create_time": 1394636235,
          "order_express_price": 0,
          "buyer_openid": "oDF3iY17NsDAW4UP2qzJXPsz1S9Q",
          "buyer_nick": "likeacat",
          "receiver_name": "张三",
          "receiver_province": "广东省",
          "receiver_city": "广州市",
          "receiver_address": "华景路一号南方通信大厦5楼",
          "receiver_mobile": "123456",
          "receiver_phone": "123456",
          "product_id": "pDF3iYx7KDQVGzB7kDg6Tge5OKFo",
          "product_name": "《教爸爸妈妈用微信》",
          "product_price": 1,
          "product_sku": "1075741873:1079742377",
          "product_count": 1,
          "product_img": "http://mmbiz.qpic.cn/mmbiz/4whpV1VZl2icND8WwMThBEcehjh
Dv2icY4GrDSG5RLM3B2qd9kOicWGVJcsAhvXfibhWRNoGOvCfMC33G9z5yQr2Qw/0",
          "delivery_id": "1900659372473",
          "delivery_company": "059Yunda",
             "trans_id": "1900000109201404103172199813"
        }
    ]
}
```

表6-4 订单查询接口返回内容参数说明

字段	说明
errcode	错误码
errmsg	错误信息
order_list	所有订单集合（字段说明详见根据订单 ID 获取订单详情）

为了在微信中实现订单查询，需要先将一个菜单设置为“订单查询”按钮，该菜单的类型为“click”，key为“WDDD”。

菜单的实现代码如下所示：

```
require_once('weixin.class.php');
$weixin = new class_weixin();

$button[] = array('type' => "view",
```

```
                  'name' => urlencode("微信小店"),
                  'url'  => "http://mp.weixin.qq.com/bizmall/mallshelf?id=&t=mall/list&biz=MzANDQxNDUwNQ==&shelf_id=1&showwxpaytitle=1#wechat_redirect",
                 );
$button[] = array('name' => urlencode("更多"),
                  'sub_button' => array(array('type' => "click",
                                              'name' => urlencode("我的订单"),
                                              'key'  => urlencode("WDDD")
                                              ),
                                        )
                  );
$menu = urldecode(json_encode(array('button' => $button)));
var_dump($weixin->create_menu($menu));
```

当用户单击“我的订单”按钮时，微信接口将接收到这一单击事件通知，并且调用微信小店的订单查询接口来查询当前用户的订单信息，实现代码如下所示。

```
//接收事件消息
private function receiveEvent($object)
{
    $content = "";
    switch ($object->Event)
    {
        case "subscribe":
            $content = "欢迎关注方倍工作室";
            break;
        case "CLICK":
            switch ($object->EventKey)
            {
                case "WDDD":
                    require_once('weixin.class.php');
                    $weixin = new class_weixin();
                    $openid = strval($object->FromUserName);
                    $orderArr = $weixin->get_detail_by_filter("{}");
                    if ($orderArr["errcode"] == -1){
                        $weixin->send_custom_message($openid, "text", "系统繁忙，请稍后再试！");
                    }
                    else if (count($orderArr["order_list"]) == 0){
                        $weixin->send_custom_message($openid, "text", "没有查询到订单记录！");
                    }else{
                        $data = array();
                        $data[] = array("title"=>urlencode("我的订单"), "description"=>"", "picurl"=>"", "url" =>"");
                        foreach ($orderArr["order_list"] as $index => $item){
                            if($item["buyer_openid"] == $openid){
                                $title = "编号：".$item["order_id"]."\n时间：".date("Y-m-d H:i:s",$item["order_create_time"])
                                        ."\n名称：".$item["product_name"]."\n总价：￥".($item["product_price"] / 100)." × ".$item["product_count"]." + ￥".($item["order_express_price"] / 100)." = ￥".($item["order_total_price"] / 100);
                                switch ($item["order_status"])
                                {
                                    case 2:
                                      $orderstatus = "待发货";
                                      break;
                                    case 3:
                                      $orderstatus = "已发货";
                                      break;
                                    case 5:
                                      $orderstatus = "已完成";
                                      break;
                                    case 8:
                                      $orderstatus = "维权中";
                                      break;
                                    default:
                                      $orderstatus = "未知状态码".$item["order_status"];
                                      break;
                                }
                                $title .= "\n状态：".$orderstatus;
                                $url = "";
                                if ($item["order_status"] == 3 && !empty($item["delivery_company"])){
                                    switch ($item["delivery_company"])
                                    {
                                        case "Fsearch_code":
                                          $expressName = "邮政EMS";
                                          break;
                                        case "002shentong":
                                          $expressName = "申通快递";
                                          break;
                                        case "066zhongtong":
                                          $expressName = "中通速递";
                                          break;
                                        case "056yuantong":
                                          $expressName = "圆通速递";
                                          break;
                                        case "042tiantian":
                                          $expressName = "天天快递";
                                          break;
                                        case "003shunfeng":
                                          $expressName = "顺丰速运";
                                          break;
                                        case "059Yunda":
                                          $expressName = "韵达快运";
                                          break;
                                        case "064zhaijisong":
                                          $expressName = "宅急送";
                                          break;
                                        case "020huitong":
                                          $expressName = "汇通快运";
                                          break;
                                        case "zj001yixun":
                                          $expressName = "易迅快递";
                                          break;
                                        default:
                                          $expressName = "未知物流公司，ID：".$item["delivery_company"];
                                          break;
                                    }
                                    $title .= "\n物流：".$expressName." ".$item["delivery_id"];
                                    if(preg_match("/^\d{3}[A-Za-z]{2,10}$/",$item["delivery_company"])){
                                        $companyEn = trim(substr($item["delivery_company"],3,strlen($item["delivery_company"])));
                                        $url = "http://m.kuaidi100.com/result.jsp?com=".strtolower($companyEn)."&nu=".$item["delivery_id"];
                                    }
                                }
                                $data[] = array("title"=>urlencode($title), "description"=>"", "picurl"=>"", "url" =>$url);
                            }
                            if (count($data) >=9){break;}
                        }

                        if (count($data) == 1){
                            $result = $weixin->send_custom_message($openid, "text", "没有查询到你的订单记录！");
                        }else{
                            $result = $weixin->send_custom_message($openid, "news", $data);
                        }
                    }
                    $content = "";
                    break;
                default:
                    $content = "空菜单响应！";
```

```
108                     break;
109             }
110             break;
111     }
112     if(is_array($content)){
113         if (isset($content[0])){
114             $result = $this->transmitNews($object, $content);
115         }else if (isset($content['MusicUrl'])){
116             $result = $this->transmitMusic($object, $content);
117         }
118     }else{
119         $result = $this->transmitText($object, $content);
120     }
121     return $result;
122 }
```

上述代码中和订单查询通知相关部分简要说明如下：

第13行：判断是否收到订单查询事件通知。

第14行~第17行：引用微信小店SDK，获取当前的所有订单信息。

第18行~第23行：用于查询订单接口的异常判断。

第24行~第96行：遍历订单详细中的所有订单列表，将当前用户的订单内容填充到微信图文消息中。

第98行~第102行：使用客服接口将订单详情通过图文消息方式发送。

最后，我的订单查询实现如图6-28所示。

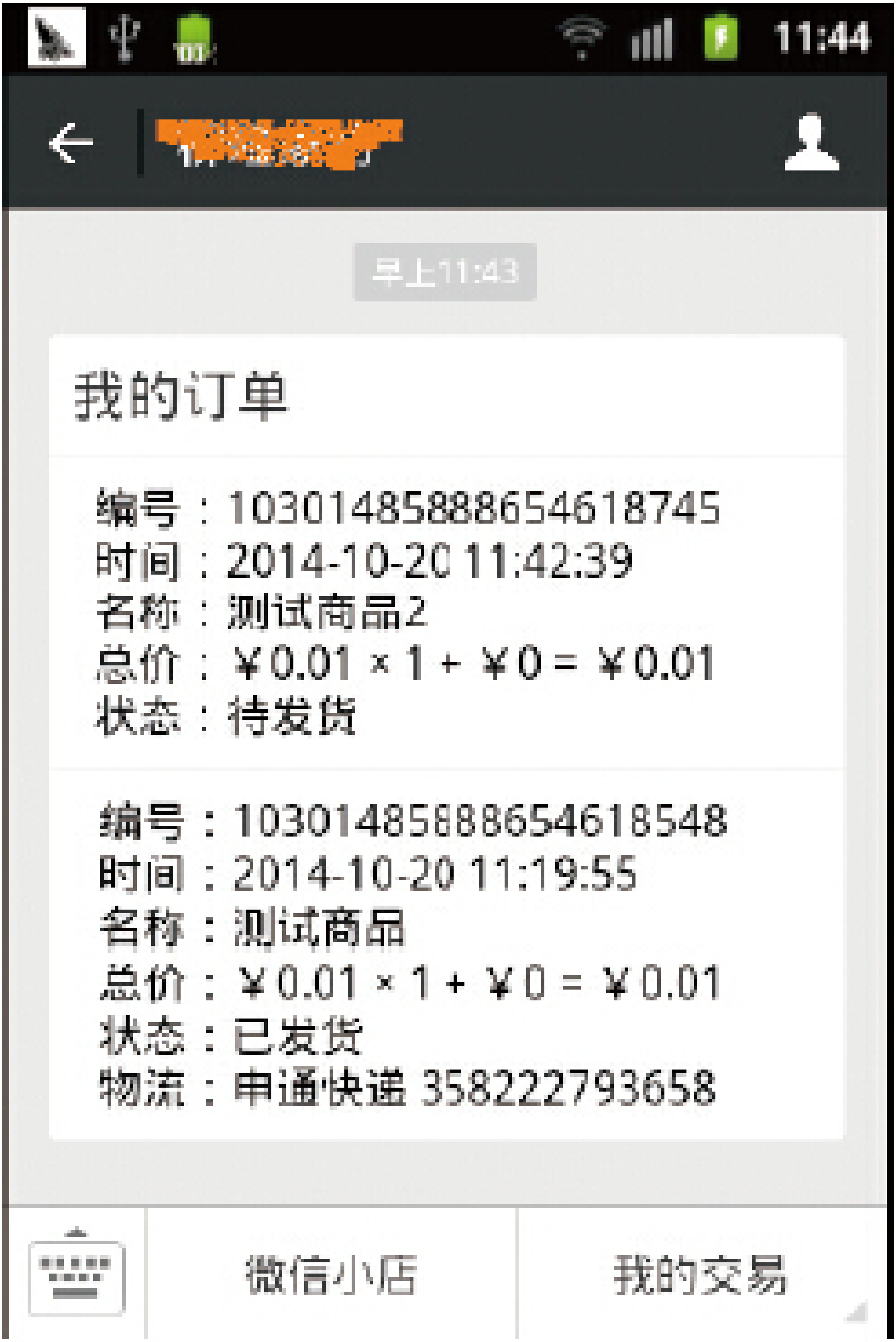

11:44
早上11:43
我的订单
编号：1030148588886546618745
时间：2014-10-20 11:42:39
名称：测试商品2
总价：￥0.01 × 1 + ￥0 = ￥0.01
状态：待发货
编号：1030148588886546618548
时间：2014-10-20 11:19:55
名称：测试商品
总价：￥0.01 × 1 + ￥0 = ￥0.01
状态：已发货
物流：申通快递 358222793658
微信小店
我的交易

图6-28　我的订单查询结果

6.2.4　快递物流查询

发货后，快递物流查询是一个最常用的功能需求。我们使用爱查快递的查询接口来实现快递查询功能。

爱查快递已集成国内外近百家常见快递/物流/COD公司的单号和网点查询功能，如EMS快递、顺丰快递、申通快递、圆通快递、中通快递、韵达快递、天天快递、汇通快递、宅急送快递、CCES、华宇物流、速尔快递、优速快递等，我们使用爱查快递来作为我们的快递查询接口，它的接口申请地址为：http://api.ickd.cn/users/。

使用邮箱在爱查快递网站上注册一个新用户，然后登录，登录后如图6-29所示。

图6-29　爱查快递API管理中心

左侧菜单目录中的“API申请”中包括“申请正式API”和“申请临时API”。如果需要长期稳定的接口，请选择申请正式API。这里以申请临时API为例，申请成功后界面如图6-30所示。

图6-30　爱查快递API基本信息

爱查快递的接口请求地址如下：

```
http://api.ickd.cn/?id=[]&secret=[]&com=[]&nu=[]&type=[]&encode=[]&ord=[]&lang=[]
```

该接口相关参数说明如表6-5所示.

表6-5　爱查快递接口参数说明

字段	是否必须	说明
com	必须	快递公司代码（英文）
nu	必须	快递单号，长度必须大于 5 位
id	必须	授权 KEY，纯数字型
secret	必选	一个小写的字符串
type	可选	返回结果类型，值分别为 html|json（默认）|text|xml
encode	可选	gbk（默认）|utf8
ord	可选	asc（默认）|desc，返回结果排序
lang	可选	en 返回英文结果，目前仅支持部分快递（EMS、顺丰、DHL）

爱查快递目前支持的快递公司列表如下：

```
AAE快递,安捷快递,安信达快递,Aramex国际快递,巴伦支,宝通达,成都奔腾国际快递,CCES快递,长通物流,程光快递,城际快递,城市100,传喜快递,传志快递,出口易物流,CityLinkExpress,东方快递,城市之星,大田物流,大洋
```

使用爱查快递接口，查询顺丰快递，单号113819507900，URL构造如下所示：

```
http://api.ickd.cn/?id=103223&secret=5d91adb21d97d109130d4671369910f4&com=shunfeng&nu=113819507900&type=json&encode=utf8
```

该接口返回结果如下：


```
{
    "status": "1",
    "message": "",
    "errCode": "0",
    "data": [
        {
            "time": "2014-02-15 11:00",
            "context": "已收件"
        },
        {
            "time": "2014-02-15 12:19",
            "context": "快件在东莞,准备送往下一站东莞集散中心"
        },
        {
            "time": "2014-02-16 01:40",
            "context": "快件在东莞集散中心,准备送往下一站"
        }
    ],
    "html": "",
    "mailNo": "113819507900",
```

```
    "expTextName": "顺丰快递",
    "expSpellName": "shunfeng",
    "update": "1392549644",
    "cache": "0",
    "ord": "ASC",
    "tel": "4008-111-111"
}
```

上述返回结果字段说明如表6-6所示。

表6-6　爱查快递接口返回字段说明

字段	类型	说明
status	int	查询结果状态，0|1|2|3|4，0 表示查询失败，1- 正常，2- 派送中，3- 已签收，4- 退回，5- 其他问题
errCode	int	错误代码，0- 无错误，1- 单号不存在，2- 验证码错误，3- 链接查询服务器失败，4- 程序内部错误，5- 程序执行错误，6- 快递单号格式错误，7- 快递公司错误，10- 未知错误，20-API 错误，21-API 被禁用，22-API 查询量耗尽
message	string	错误消息
data	array	进度
html	string	其他 HTML，该字段不一定存在
mailNo	string	快递单号
expSpellName	string	快递公司英文代码
expTextName	string	快递公司中文名
update	int	最后更新时间（unix 时间戳）
cache	int	缓存时间，当前时间与 update 之间的差值，单位为：秒
ord	string	排序，ASC|DESC
tel	string	客服电话，该字段不一定存在

根据上述接口，我们编写快递查询代码如下：

```
function getIckdExpressInfo($companyEn, $number)
{
    $appid = "103223";
    $secret = "5d91adb21d97d109130d4671369910f4";
    $url= "http://api.ickd.cn/?id=".$appid."&secret=".$secret."&com=".$companyEn."
&nu=".$number."&type=json&encode=utf8";
    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($ch);
    if(curl_errno($ch))
    { echo 'CURL ERROR Code: '.curl_errno($ch).', reason: '.curl_error($ch);}
    curl_close($ch);
    $AllInfo = json_decode($output, true);
    if($AllInfo['message'] != "" ){
        return $AllInfo['message'];
    }else{
        $result = "";
        foreach ($AllInfo["data"] as $singleStep)
        {
            $result .= $singleStep["time"]." ".$singleStep["context"]."\n";
        }
        return trim($result);
    }
}
```

快递公司众多，而我们需要都能查询得到。在查询的时候，一般使用快递公司名再加上单号的方法，而快递公司名称既有中文名称也有英文名称，根据这一原则，我们使用正则表达式来判断，相应代码编写如下：

```
private function receiveText($object)
{
    $keyword = trim($object->Content);
    if (preg_match("/[\x{4e00}-\x{9fa5}]{2,4}\s{0,}\w{8,15}/u",$keyword) ||
        preg_match("/[A-Za-z]{3,9}\s{0,}\w{8,15}/u",$keyword)
        ){
        include("express.php");
        $content = getExpressInfo($keyword);
    }else{
        $content = "不是快递查询格式？";
    }
    $result = $this->transmitText($object, $content);
    return $result;
}
```

上述代码中，判断用户发送过来的是不是满足这样两种规则：

· 2~4位汉字+0~N个空格+8~15位数字或字母

· 3~8位字母+0~N个空格+8~15位数字或字母

如果满足以上两者中的任何一种，则认为是快递查询请求格式。进入快递查询请求函数，代码如下：

```
 1 function getExpressInfo($keyword)
 2 {
 3     $expresses = array(
 4         "AAE"=>"aae","安捷"=>"anjie","安信达"=>"anxinda","ARAMEX"=>"aramex","巴伦支"=>"balunzhi","宝通达"=>"baotongda","成都奔腾"=>"benteng","CCES"=>"cces","长通"=>"changtong","
 5     );
 6     $contentStr = "";
 7     for($len = 3; $len <= 12; $len++)
 8     {
 9         if (array_key_exists(substr($keyword, 0, $len), $expresses) || array_
key_exists(strtoupper(substr($keyword, 0, $len)), $expresses)){
10             if (preg_match("/[a-zA-Z]/",substr($keyword, 0, $len))){
11                 $companyEn = $expresses[strtoupper(substr($keyword, 0, $len))];
12             }else{
13                 $companyEn = $expresses[substr($keyword, 0, $len)];
14             }
15             $number = trim(substr($keyword, $len, strlen($keyword)));
16             if (!preg_match("/^\w{8,14}$/",$number)){
17                 return "请发送'快递名称'+'单号'，不要加快递两个字。例如"汇通210157777433"";
18             }
19             $contentStr = getIckdExpressInfo($companyEn, $number);
20             return $contentStr;
21         }
22     }
23     if ($contentStr == ""){
24         $contentStr = "没有匹配到快递公司！";
25         return $contentStr;
26     }
27 }
```

上述代码解读如下：

第3行~第5行：定义快递公司名称及相应的查询代码。在这里，我们把所有包含英文的名称都转成大写，以便后续方便匹配。

第7行~第9行：使用循环遍历长度的方式，匹配快递公司名称。快递公司名称最短为3个字母（3字节），最长不超过4个汉字（utf-8编码下12字节）所以长度值为3~12。匹配的方法同时考虑中文或者字母的方式。对于字母，将其强制转换成大写形式。

第10行~第14行：根据快递公司名称获取相应的快递公司英文代码。若含有英文字母，获取之前将其转换成大写形式。

第15行~第16行：剥离出快递单号，并且判断单号是否由数字组字组成。如果不是，则匹配快递公司名称失败（遗漏汉字到单号中或者用户多输出了数组中不包括的汉字）。

第19行~第20行：根据取得的快递公司名称和名称调用爱查快递查询接口查询快递的当前进度，并且返回。

第23行~第26行：如果上述循环遍历没有找到快递公司名称，则返回匹配快递公司名称失败。

需要说明的是：由于快递公司名称的不规则性太强，上述方法仍有可能存在无法匹配到快递公司名称和单号的情况。这种情况下，需要更进一步分析出错原因，找出代码中疏漏的地方。

上述快递查询代码最终实现效果如图6-31所示。

图6-31　快递查询实现效果

6.2.5 维权功能接入

微信小店中，如果需要实现用户退货退款等功能，则需要在菜单中加入“维权”按钮，让用户可以到达维权页面。加入方法很简单，给对应名称为“我要维权”的菜单设置链接（view事件），URL为

```
https://mp.weixin.qq.com/payfb/payfeedbackindex?appid=wx830f2de3fabcdefg#wechat_webview_type=1&wechat_redirect
```

把其中的appid换成当前公众账号的即可。

菜单中的实现代码如下所示。

```
require_once('weixin.class.php');
$weixin = new class_weixin();

$button[] = array('type' => "view",
                  'name' => urlencode("微信小店"),
                  'url'  => "http://mp.weixin.qq.com/bizmall/mallshelf?id=&t=mall/list&biz=MzANDQxNDUwNQ==&shelf_id=1&showwxpaytitle=1#wechat_redirect",
                 );
$button[] = array('name' => urlencode("更多"),
                  'sub_button' => array(array('type' => "click",
                                              'name' => urlencode("我的订单"),
                                              'key'  => urlencode("WDDD")
                                              ),
                                        array('type' => "view",
                                              'name' => urlencode("维权"),
                                              'url'  => "https://mp.weixin.qq.com/payfb/payfeedbackindex?appid=wx830f2de3fabcdefg#wechat_webview_type=1&wechat_redirect"
                                             ),
                                        )
                  );
$menu = urldecode(json_encode(array('button' => $button)));
var_dump($weixin->create_menu($menu));
```

上述维权代码实现效果如图6-32所示。

图6-32　维权接入效果

6.3　本章小结

本章完整地介绍了微信小店的搭建方式，并对微信小店二次开发的付款通知、订单查询、物流查询、维权功能进行了讲解，这些功能对微信小店进行了扩展，使其功能更加完善。

第7章　生活类应用开发

微信公众平台作为一个能支持自动回复的平台，特别适合查询类应用的开发。本章将向读者介绍常用的生活类应用如何在微信公众平台上实现。本章内容将包括：天气预报、交通信息查询、中英翻译、空气质量、股票行情及分析、苹果产品信息查询，历史上的今天，附近搜索。

7.1　天气预报

俗话说，天有不测风云。人们外出都需要关心天气，以决定需要如何穿衣及是否需要带雨具。本节将介绍如何在微信公众平台上实现天气预报功能。

7.1.1　中国气象局

SmartWeatherAPI（简称“SWA接口”）是中国气象局推出的手机气象服务客户端API，供Android、iOS、WindowsPhone等各类智能手机气象服务客户端快速开发使用。API接口提供包括地理定位、天气实况、天气预报、天气预警、天气资讯、消息推送服务、用户注册登录、增值服务、集成第三方分享、用户反馈收集等各类用于手机气象服务客户端开发的相关功能，是移动互联网气象应用服务的重要基础设施，是中国气象局开展创新型公共气象服务的重要技术支撑平台。

API接口旨在提供涵盖国内外超过10000个城市（未来将扩展于全球任意地点）的气象服务信息，包括天气实况、7天时长（3、6、12小时时效）的天气预报、生活气象指数、城市天气预警，历史气候数据、天气资讯和气象视频等内容，以及由中国气象局各省、市、县气象局提供的更具针对性的本地化增值服务、专业专项服务等内容，且各类增值服务与专业专项内容可直接与用户权限挂钩，便于开展更为精细化、多样化和本地化的特色气象服务。同时API接口还集成了消息推送服务、SNS、用户反馈收集等功能，便于气象服务类手机客户端的快速应用开发。

用户使用API接口，须得到中国气象局公共气象服务中心的官方授权，获取Appkey即可使用，具体授权申请方式和流程，请查询如下网址：http://smart.weather.com.cn，或者垂询气象服务热线，电话：400-6000-121。

得到接口授权后，将可获取API接口的访问接口，包括appid和private_key。

SWA接口数据目前主要包括实况、指数、常规预报等数据内容。接口地址为

```
http://webapi.weather.com.cn/data/
```

相关参数如表7-1所示。

表7-1　SWA接口参数说明

参数名称	参数含义	参数样例
areaid	区域 id	101010100
type	数据类型	实况 :observe 指数 :index 常规预报 :forecast3d
date	客户端日期	按照格式 yyyyMMddHHmm 获取客户端当前时间
appid	固定分配的型号标识	传递参数时：截取 appid 的前 6 位 生成公钥时：取完整的 appid
key	令牌	由公钥 (public_key) 和私钥 (private_key) 通过固定算法加密生成

其中，public_key为不包含key在内的完整URL的其他部分（appid为完整appid）。

示例如下：

```
http://webapi.weather.com.cn/data/?areaid=101010100&type=forecast&date=201211281030&appid=cf2d61521456sads
```

key加密后通过urlencode对其编码后传参，其算法如下：

```
key=base64_encode(hash_hmac('sha1',$public_key,$private_key,TRUE));
```

获取实况信息的URL如下所示：

```
http://webapi.weather.com.cn/data/?areaid=101010100&type=observe&date=201403100032&appid=123456&key=WIElp125RskWgo6%2BSGnotpi%2BERY%3D
```

该接口以JSON格式列出当前城市的实时天气状况。如下所示：


```
{
    "l": {
        "l1": "8",
        "l2": "37",
        "l3": "2",
        "l4": "3",
        "l7": "14:15"
    }
}
```


上述数据参数说明如表7-2所示。

表7-2　实况天气数据参数说明

参数名称	参数样例	参数含义
l1	8	当前温度（摄氏度）
l2	38	当前湿度（单位 %）
l3	6	当前风力（单位是级）
l4	3	当前风向编号
l7	11:00	实况发布时间

获取指数信息的URL如下所示。

```
http://webapi.weather.com.cn/data/?areaid=101010100&type=index&date=201403100044&appid=123456&key=rFXQDnrcNUO4FmCN%2BhjoD91G7Qw%3D
```

该接口以JSON格式列出天气指数状况，如下所示：


```
{
"i": [
        {
            "i1": "ct",
            "i2": "穿衣指数",
            "i3": "",
            "i4": "较冷",
            "i5": "建议着厚外套加毛衣等服装。年老体弱者宜着大衣、呢外套加羊毛衫。"
        },
        {
            "i1": "zs",
            "i2": "中暑指数",
            "i3": "",
            "i4": "无",
            "i5": "温度不高，其他各项气象条件适宜，中暑机率极低。",
            "i6": "无",
            "i7": "温度不高，其他各项气象条件适宜，中暑机率极低。",
            "i8": "无",
            "i9": "温度不高，其他各项气象条件适宜，中暑机率极低。",
            "i10": ""
        }
    ]
}
```


上述数据参数说明如表7-3所示。

表7-3 指数天气数据参数说明

参数名称	参数含义	参数样例
i1	指数简称	ct
i2	指数中文名称	穿衣指数
i3	指数中文别名	
i4	指数级别	热
i5	指数内容	天气热，建议着短裙、短裤、短薄外套、T恤等夏季服装

获取3日预报数据的URL如下所示：

```
http://webapi.weather.com.cn/data/?areaid=101010100&type=forecast3d&date=201403100048&appid=123456&key=ivQAc8XOBdumpYM3xIy%2BmSB3Rxs%3D
```

该接口以JSON格式列出3天24小时常规预报数据。如下所示：


```
{
    "c": {
        "c1": "101010100",
        "c2": "beijing",
        "c3": "北京",
        "c4": "beijing",
        "c5": "北京",
        "c6": "beijing",
        "c7": "北京",
        "c8": "china",
        "c9": "中国",
        "c10": "1",
        "c11": "010",
        "c12": "100000",
        "c13": "116.391",
        "c14": "39.904",
        "c15": "33",
        "c16": "AZ9010",
        "c17": "+8"
    },
    "f": {
        "f1": [
            {
                "fa": "00",
                "fb": "01",
                "fc": "11",
                "fd": "2",
                "fe": "0",
                "ff": "0",
                "fg": "0",
                "fh": "0",
                "fi": "06:34|18:15"
            },
            {
                "fa": "02",
                "fb": "02",
                "fc": "11",
                "fd": "3",
                "fe": "0",
                "ff": "8",
                "fg": "0",
                "fh": "1",
                "fi": "06:32|18:16"
            },
            {
                "fa": "00",
                "fb": "00",
                "fc": "10",
                "fd": "-2",
                "fe": "0",
                "ff": "0",
                "fg": "0",
                "fh": "0",
                "fi": "06:30|18:17"
            }
        ],
        "f0": "201403101100"
    }
}
```


上述数据参数说明如表7-4所示。

表7-4 常规预报数据参数说明

参数名称	参数含义	参数样例	参数名称	参数含义	参数样例
c1	区域	ID	c14	纬度	39.904
c2	城市英文名	beijing	c15	海拔	33
c3	城市中文名	北京	c16	雷达站号	AZ9010
c4	城市所在市英文名	beijing	f0	预报发布时间	201203061100
c5	城市所在市中文名	北京	fa	白天天气现象编号	01
c6	城市所在省英文名	beijing	fb	晚上天气现象编号	01
c7	城市所在省中文名	北京	fc	白天天气温度（摄氏度）	11
c8	城市所在国家英文名	china	fd	晚上天气温度（摄氏度）	0
c9	城市所在国家中文名	中国	fe	白天风向编号	4
c10	城市级别	1	ff	晚上风向编号	4
c11	城市区号	010	fg	白天风力编号	1
c12	邮编	100000	fh	晚上风力编号	0
c13	经度	116.391	fi	日出日落时间（中间用 | 分割）	06:44|18:21

表7-4中的天气现象编码表说明如表7-5所示。

表7-5　天气现象编码表

编号	中文名称	英文名称	编号	中文名称	英文名称
0	晴	Sunny	17	暴雪	Snowstorm
1	多云	Cloudy	18	雾	Foggy
2	阴	Overcast	19	冻雨	Ice rain
3	阵雨	Shower	20	沙尘暴	Duststorm
4	雷阵雨	Thundershower	21	小到中雨	Light to moderate rain
5	雷阵雨伴有冰雹	Thundershower with hai	22	中到大雨	Moderate to heavy rain
6	雨夹雪	Sleet	23	大到暴雨	Heavy rain to storm
7	小雨	Light rain	24	暴雨到大暴雨	Storm to heavy storm
8	中雨	Moderate rain	25	大暴雨到特大暴雨	Heavy to severe storm
9	大雨	Heavy rain	26	小到中雪	Light to moderate snow
10	暴雨	Storm	27	中到大雪	Moderate to heavy snow
11	大暴雨	Heavy storm	28	大到暴雪	Heavy snow to snowstorm
12	特大暴雨	Severe storm	29	浮尘	Dust
13	阵雪	Snow flurry	30	扬沙	Sand
14	小雪	Light snow	31	强沙尘暴	Sandstorm
15	中雪	Moderate snow	53	霾	Haze
16	大雪	Heavy snow	99	无	Unknown

表7-4中的风向风力编码表说明如表7-6所示。

表7-6　风向风力编码表

风向编号	中文名称	英文名称	风力编号	中文名称	英文名称
0	无持续风向	Nowind	0	微风	<10m/h
1	东北风	Northeast	1	3-4 级	10 ～ 17m/h
2	东风	East	2	4-5 级	17 ～ 25m/h
3	东南风	Southeast	3	5-6 级	25 ～ 34m/h
4	南风	South	4	6-7 级	34 ～ 43m/h
5	西南风	Southwest	5	7-8 级	43 ～ 54m/h
6	西风	West	6	8-9 级	54 ～ 65m/h
7	西北风	Northwest	7	9-10 级	65 ～ 77m/h
8	北风	North	8	10-11 级	77 ～ 89m/h
9	旋转风	Whirl wind	9	11-12 级	89 ～ 102m/h

7.1.2 区域编码表

获取到授权以后，SWA同时会提供城市区域编码表。需要将城市区域编码列表导入数据库中。

SAE提供了MySQL数据库服务，开发者可以使用phpMyAdmin来进行数据管理，如图7-1所示。

图7-1　SAE的MySQL服务

单击“管理MySQL”按钮，输入安全密码后，就进入phpMyAdmin管理界面，如图7-2所示。

图7-2 phpMyAdmin管理界面

SAE已经为每个应用分配了一个数据库，上例中数据库名称为“app_cctv15”。

在phpMyAdmin中创建数据表有3种方法：

· 在“结构”页面中，手工填写表名称、字段名称及各字段属性。

· 在“SQL”页面中，编写SQL语句命令来创建。

· 在“导入”页面中，使用导入功能将建表文件导入，自动创建表及字段。

本书已提供建表文件，这里使用导入的方式来创建。读者若对MySQL及phpMyAdmin比较生疏，可以自行查阅相关资料深入了解。

单击“导入”链接，在页面中选择要导入的文件为weather.sql，再单击“执行”按钮，如图7-3所示。

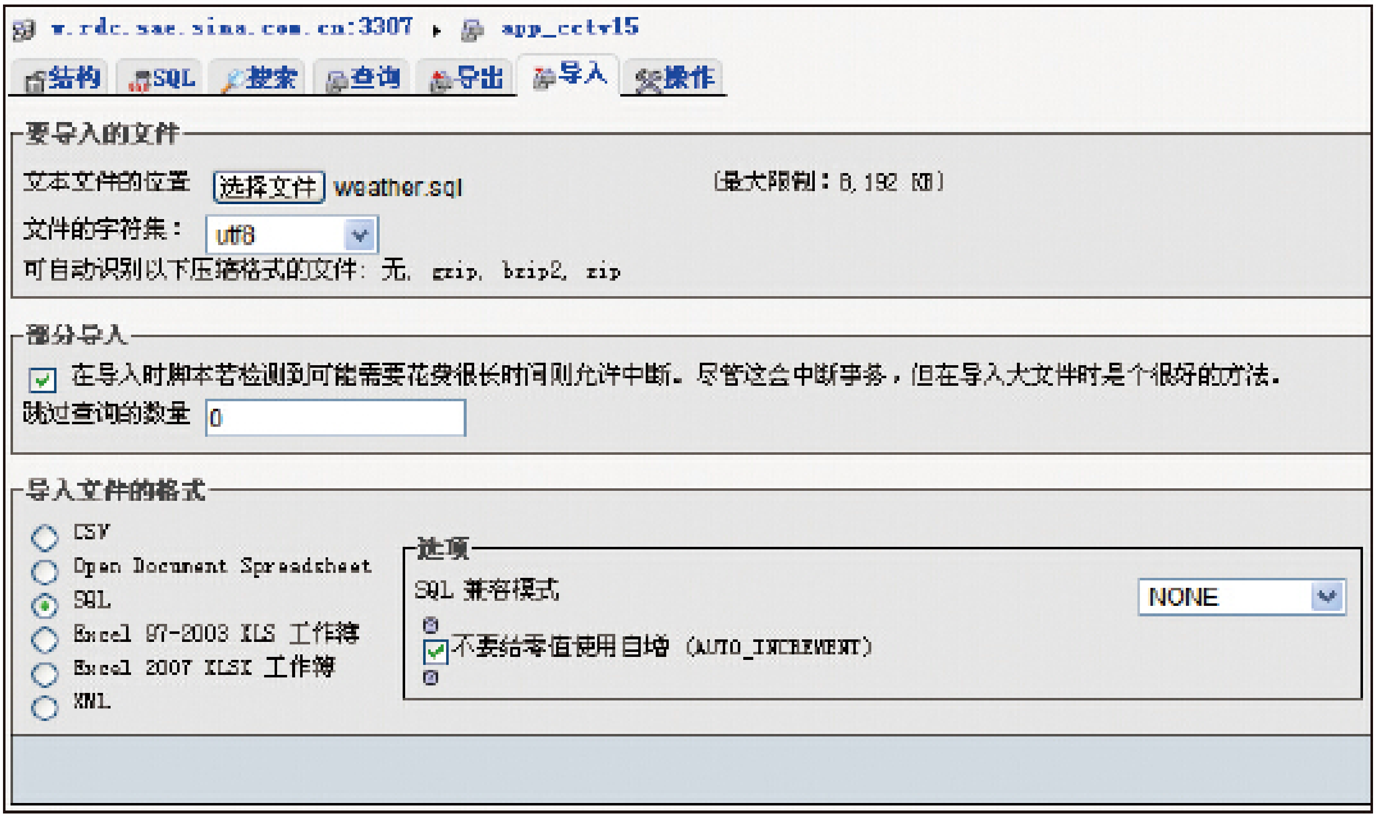


图7-3 phpMyAdmin文件导入

导入成功后，在“浏览”页面中可以看到数据记录，如图7-4所示。

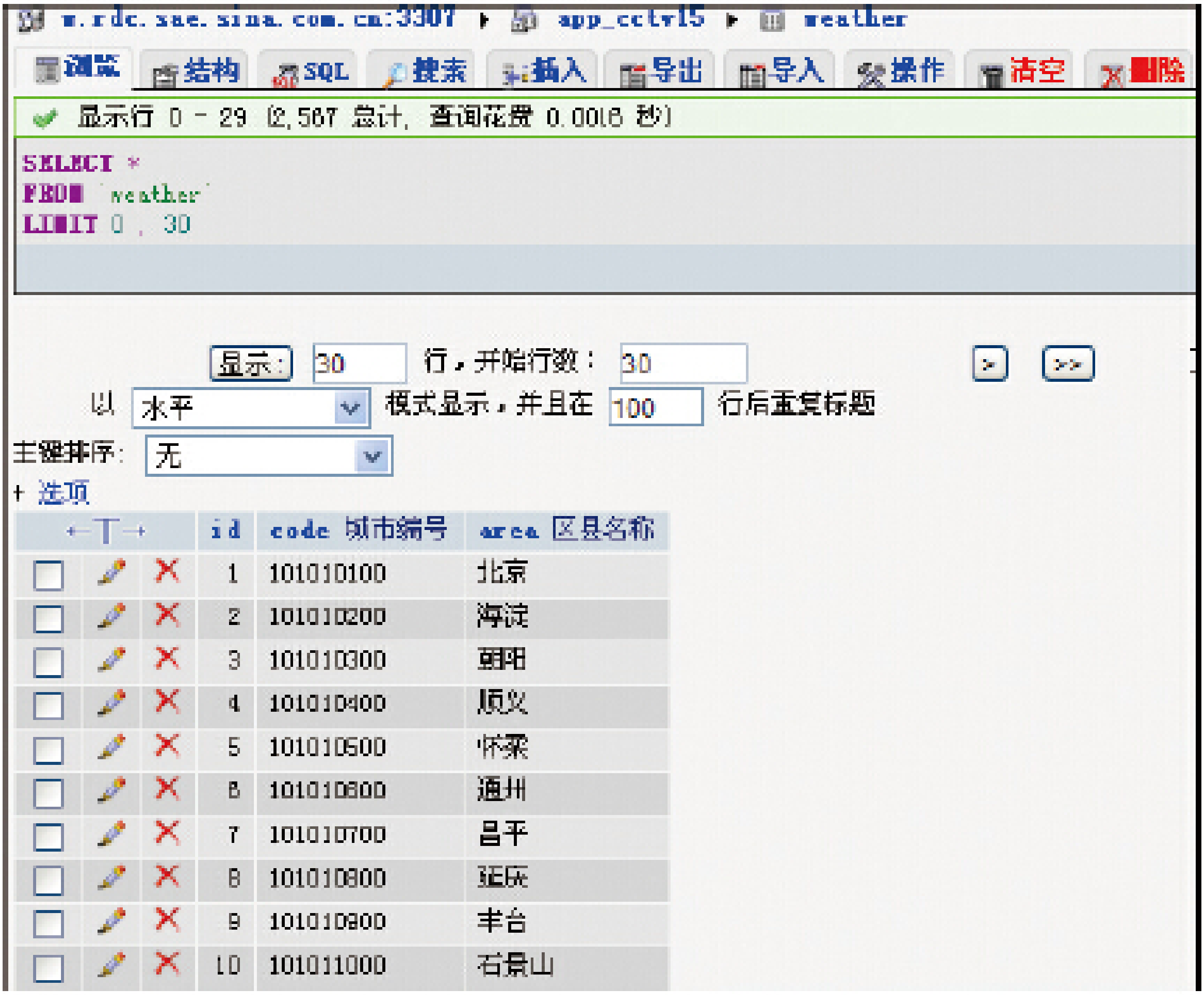


图7-4 phpMyAdmin文件导入

7.1.3 开发实现

当用户发送城市名称以后，我们需要查询到该城市的城市编码，下面以PHP操作MySQL数据库为例，展示了从数据库中查询城市编码的方法：

```
 1 function fromKeywordToCode($keyword)
 2 {
 3     Global $mysql_host;
 4     Global $mysql_host_s;
 5     Global $mysql_port;
 6     Global $mysql_user;
 7     Global $mysql_password;
 8     Global $mysql_database;
 9
10     $mysql_table = "weather";
11    $mysql_state = "SELECT * FROM `".$mysql_table."` WHERE `area` LIKE  '%".
$keyword."%' LIMIT 0 , 1";
12     $con = @mysql_connect($mysql_host.':'.$mysql_port, $mysql_user, $mysql_password, true);
13     if (!$con){
14         die('Could not connect: ' . mysql_error());
15     }
16     mysql_query("SET NAMES 'UTF8'");
17     mysql_select_db($mysql_database, $con);
18     $result = mysql_query($mysql_state);
19     $citycode = "";
20     while($row = mysql_fetch_array($result))
21     {
22         $citycode = $row['code'];
23         break;
24     }
25     mysql_close($con);
26     return $citycode;
27 }
```

上述代码解读如下：

第3行~第8行：定义了数据库的环境，这些环境变量包括用户名、密码、主库域名、从库域名、端口、数据库名。

第10行~第11行：构造了查执行查询的SQL语句。从表“weather”中查询指定城市名称的相关内容。

第12行~第15行：进行数据库连接。如果连接失败，将执行“die”部分。

第16行：设置字符集为“UTF8”，以便正常显示中文。

第17行：设置活动的MySQL数据库。

第18行：使用mysql_query（）函数向MySQL发送命令。

第19行~第24行：将查询到的结果存在变量中。

第25行：关闭数据库连接。

第26行：返回查询到的城市编码。

我们使用cURL获取天气接口的数据，获取代码如下：

```
 1 function getWeatherData($areaid, $type)
 2 {
 3     $appid = "123456****abcdef";
 4     $private_key = "******_SmartWeatherAPI_*******";
 5
 6     $urls = 'http://webapi.weather.com.cn/data/?areaid='.$areaid.' &type='.
$type.'&date='.date("YmdHi").'&appid=';
 7     $public_key = $urls.$appid;
 8     $key = urlencode(base64_encode(hash_hmac('sha1', $public_key, $private_key,
true)));
 9     $url = $urls.substr($appid,0,6).'&key='.$key;
10
11     $ch = curl_init();
12     curl_setopt($ch, CURLOPT_URL, $url);
13     curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
14     $output = curl_exec($ch);
15     curl_close($ch);
16     return $output;
17 }
```

上述代码解读如下：

第3行~第4行：定义了SWA接口。

第6行：定义一个urls。将用于生成public_key以及请求接口url。

第7行：拼装public_key。

第8行：生成key。

第9行：生成要请求数据的url。

下面结合上面的代码，介绍cURL的基本用法：

1）初始化会话，使用curl_init（），如上述代码中第11行所示。

2）为一个curl设置会话参数，使用curl_setopt（），如上述代码中第12行~第13行所示。代码中的两个选项含义如下：

```
CURLOPT_URL：要获取的URL地址。
CURLOPT_RETURNTRANSFER：是否返回原生的输出。
```

cURL中还有很多可以设置的选项，读者可以参考PHP官方网站关于curl_setopt的介绍，地址为：

```
http://cn2.php.net/manual/zh/function.curl-setopt.php。
```

3）执行curl会话，使用curl_exec，如上述代码中第14行所示。

4）关闭curl会话，如上述代码中第15行所示。

由于气象数据有三种类型，我们根据需求获取相应的气象数据结果，再将天气数据组装成微信的图文消息，下面是获取实时天气、指数、未来3日天气的代码：

```
function getWeatherInfo($cityName)
{
    if ($cityName == ""){
        return "发送天气+城市，例如'天气深圳'";
    }
    $citycode = fromKeywordToCode($cityName);
    if ($citycode == ""){
        return "没有该城市？";
    }

    $winddirection = array("0"=>"无持续风向","1"=>"东北风","2"=>"东风", "3"=>"东南风","4"=>"南风","5"=>"西南风","6"=>"西风","7"=>"西北风","8"=>"北风","9"=>"旋转风");
    $windpower = array("0"=>"微风","1"=>"3-4级","2"=>"4-5级","3"=>"5-6级","4"=>"6-7级","5"=>"7-8级","6"=>"8-9级","7"=>"9-10级","8"=>"10-11级", "9"=>"11-12级");
     $phenomenon = array("00"=>"晴","01"=>"多云","02"=>"阴","03"=>"阵雨", "04"=>"雷阵雨","05"=>"雷阵雨伴有冰雹","06"=>"雨夹雪","07"=>"小雨","08"=>"中雨","09"=>"大雨","10"=>"暴雨",

    $weatherArray = array();
     $weatherArray[] = array("Title" =>$cityName."天气预报", "Description" =>"", "PicUrl" =>"", "Url" =>"");

    //实况
     $liveInfo = json_decode(getWeatherData($citycode, "observe"), true);
     $weatherArray[] = array("Title" =>"【实况】温度". $liveInfo["l"]["l1"]."℃ 湿度".$liveInfo["l"]["l2"]."%% ". $winddirection[$liveInfo["l"]["l4"]].$liveInfo["l"]["l3"]."级

    //指数
     $indexInfo = json_decode(getWeatherData($citycode, "index"), true);
    $indextitle = "";
    for ($i = 0; $i < count($indexInfo["i"]); $i++) {
            $indextitle .= "【".$indexInfo["i"][$i]["i4"]."】".$indexInfo["i"][$i]["i5"]."\n";
        break;
    }
     $weatherArray[] = array("Title" =>trim($indextitle), "Description" =>"", "PicUrl" =>"", "Url" =>"");

    //3日
     $forecast3dInfo = json_decode(getWeatherData($citycode, "forecast3d"), true);
    $day3Info = $forecast3dInfo["f"]["f1"];
    $weekArray = array("日","一","二","三","四","五","六");
    for ($i = 0; $i < count($day3Info); $i++) {
            $offset = strtotime("+".((date("H") >=18)?($i+1):$i)." day");
            $forecast3dtitle = date("m月d日",$offset)." 周".$weekArray[date('w',$offset)]." ".
            $phenomenon[$day3Info[$i]["fa"]].(($day3Info[$i]["fa"] != $day3Info[$i]["fb"])?("转".$phenomenon[$day3Info[$i]["fb"]]):"")." ".
            $day3Info[$i]["fc"]."℃~".$day3Info[$i]["fd"]."℃ ".
            $winddirection[$day3Info[$i]["fe"]]." ".$windpower [$day3Info[$i]["fg"]]." ".
        "日出日落：".$day3Info[$i]["fi"];
            $picurl = "http://discuz.comli.com/weixin/swa/".((date("H") >=18 && $i ==0)?("night/".$day3Info[$i]["fb"]):("day/".$day3Info[$i]["fa"]))).".png";
            $weatherArray[] = array("Title" =>trim($forecast3dtitle), "Description" =>"", "PicUrl" =>$picurl, "Url" =>"");
    }
    return $weatherArray;
}
```

上述代码解读如下：

第3行~第9行：判断城市名及获取城市区域编码。

第11行~第13行：定义风向风力编码表及天气现象编码表。

第18行~第20行：获取实时天气数据，将其加入到多图文消息中。

第22行~第29行：获取指数天气数据，将其加入到多图文消息中。

第31行~第44行：获取未来3日天气数据，根据数据转换成相应的文字说明，同时拼接相应的图片url，并将其加入到多图文消息中。

最终，一个微信上的天气预报功能如图7-5所示。

图7-5　微信天气预报

7.2　交通信息查询

差不多人人每天都要接触到公交、火车、汽车、飞机这些交通工具，都会有需要了解的交通信息。本节将介绍如何在微信公众平台上借助第三方网站实现交通信息的查询。

7.2.1　第三方网站

交通信息是属于公共信息，已经有很多大型网站提供了相应的查看和查询功能，但这些信息的后台数据往往并不对外公开，一是因为获取成本太高，二则怕添加一个竞争对手。但好在很多第三方网站提供了手机版的查询功能。我们可以直接在微信中链接这些手机版的网站。

以下是精选出的一部分交通信息查询的手机版网站。

公交：http://m.8684.cn/bus

汽车：http://touch.trip8080.com/

火车：http://m.ctrip.com/html5/Trains/

飞机：http://touch.qunar.com/h5/flight/

城市路况：http://dp.sina.cn/dpool/tools/citytraffic/city.php

违章查询：http://app.eclicks.cn/violation2/webapp/index? appid=10

7.2.2 开发实现

我们直接在微信上回复图文消息，然后消息中链接到这些网站。代码如下所示：

```
private function receiveText($object)
 {
    $keyword = trim($object->Content);
    $content[] = array("Title" =>"交通信息","Description" =>"", "PicUrl" =>"", "Url" =>"http://m.8684.cn/bus");
    $content[] = array("Title" =>"【公交线路】\n全车公交查询", "Description" =>"", "PicUrl" =>"http://photo.candou.com/ai/114/09caed4a27c56000bb870c68ab028850", "Url" =>"http:/
    $content[] = array("Title" =>"【汽车班次】\n长途汽车班车", "Description" =>"", "PicUrl" =>"http://photo.candou.com/i/175/4951bd2a2f368cafc5ad09ff95ce591d", "Url" =>"http://
    $content[] = array("Title" =>"【火车时刻】\n火车时刻查询", "Description" =>"", "PicUrl" =>"http://photo.candou.com/ai/114/26a9407dcedda5f4a30b195f78ec3680", "Url" =>"http:/
    $content[] = array("Title" =>"【飞机票】\n机票查询", "Description" =>"", "PicUrl" =>"http://photo.candou.com/i/175/a1bd6303a8bde7da50166aa8dafb7568", "Url" =>"http://touch
    $content[] = array("Title" =>"【城市路况】\n重点城市实时路况", "Description" =>"", "PicUrl" =>"http://photo.candou.com/i/175/0d8488edf94574651d048025596a6190", "Url" =>"http
    $content[] = array("Title" =>"【违章查询】\n全国违章查询", "Description" =>"", "PicUrl" =>"http://g.hiphotos.bdimg.com/wisegame/pic/item/9e1f4134970a304eab30503cd0c8a786c81

    $result = $this->transmitNews($object, $content);
    return $result;
 }
```

实现效果如图7-6~图7-9所示。

图7-6　微信回复交通信息

图7-7 公交查询和汽车查询

图7-8　火车查询和机票查询

图7-9　实时路况和违章查询

7.3　中英翻译

在我们日常生活中大量接触到英语，当碰到不熟悉的单词或者句子时，希望能有个随时随地的英文字典进行查询。本文介绍如何在微信上开发一个中英互译的功能。

7.3.1　有道翻译

目前有很多第三方翻译接口，比如有道词典、百度等，开发人员根据这些接口可自己实现中英翻译。下面我们以网易的有道翻译接口为例，讲解如何开发中英翻译功能。

有道翻译接口申请地址如下：

```
http://fanyi.youdao.com/openapi?path=data-mode
```

在页面中填写相应的内容，再单击“申请”按钮，将得到相应的API key和keyfrom，如图7-10所示。

图7-10　申请有道翻译API

网易有道翻译接口如下：

```
http://fanyi.youdao.com/openapi.do?keyfrom=<keyfrom>&key=<key>&type=data&doctype=<doctype>&version=1.1&q=要翻译的文本
```

该接口的参数说明如表7-7所示。

表7-7　有道翻译接口参数说明

参数	含义
type	返回结果的类型，固定为 data
doctype	返回结果的数据格式，xml 或 json 或 jsonp
version	版本，当前最新版本为 1.1
q	要翻译的文本，不能超过 200 个字符，需要使用 utf-8 编码

该接口调用举例如下：

```
http://fanyi.youdao.com/openapi.do?keyfrom=txw1958&key=876842XXX&type=data&doctype=json&version=1.1&q= 你好
```

返回结果列表如下所示：

```
{
    "translation": [
        "How are you"
    ],
    "basic": {
        "phonetic": "nǐ hǎo",
        "explains": [
            "hello; hi"
        ]
    },
    "query": "你好",
    "errorCode": 0,
    "web": [{"value":["Hello","How do you do","hi"],"key":"你好"},{"value":["How are you","How Do You Do","Como Vai Voce"],"key":"你好吗"},{"value":["Teacher Kim Bong-du","My Te
}
```

7.3.2　开发实现

根据上述接口，我们编写有道翻译的代码如下：

```
 1 function getTranslateInfo($keyword)
 2 {
 3     if ($keyword == ""){
 4         return "要翻译的内容是什么";
 5     }
 6     $apihost = "http://fanyi.youdao.com/";
 7     $apimethod = "openapi.do?";
 8      $apiparams = array('keyfrom'=>"txw1958", 'key'=>"876842XXX", 'type'=>"data", 'doctype'=>"json", 'version'=>"1.1", 'q'=>$keyword);
 9     $apicallurl = $apihost.$apimethod.http_build_query($apiparams);
10
11     $ch = curl_init();
12     curl_setopt($ch, CURLOPT_URL, $apicallurl);
13     curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
14     $output = curl_exec($ch);
15     if(curl_errno($ch))
16      { echo 'CURL ERROR Code: '.curl_errno($ch).', reason: '.curl_error($ch);}
17     curl_close($ch);
18
19     $youdao = json_decode($output, true);
20     $result = "";
21     switch ($youdao['errorCode']){
22         case 0:
23             if (isset($youdao['basic'])){
24                 $result .= $youdao['basic']['phonetic']."\n";
25                 foreach ($youdao['basic']['explains'] as $value) {
26                     $result .= $value."\n";
27                 }
28             }else{
29                 $result .= $youdao['translation'][0];
30             }
31             break;
32         default:
33              $result = "系统错误：错误代码：". $errorcode;
34             break;
35     }
36     return trim($result);
37 }
```

上述代码解读如下：

第6行~第9行：构造查询请求URL。

第11行~第17行：使用cURL获取查询结果。

第19行~第36行：根据查询结果提取翻译内容。

一个查询成功的结果如图7-11所示。

22:32
返回
方倍工作室
翻译有志者事竟成
yōu zhi zhē shi jing chéng
where there is a will there is a way
翻译When someone else's happiness is your happiness, that is love.
当别人的幸福是你的幸福，那就是爱。
发送

图7-11　中英翻译

7.4　空气质量

近年来，空气污染越来越严重，城市空气质量不达标的天数也越来越多，这严重影响了人们的日常生活。本节介绍如何在微信上实现空气质量查询功能。

7.4.1　PM25.in

PM25.in是广州BestApp工作室开发的一个空气质量数据监测产品，它收录了全国190个城市，近千个监测点的实时数据（每小时更新）。数据源于国家环境保护部。开发团队将所有采集的数据进行统一处理，然后再通过API的方式提供出去。

该接口申请地址为http://pm25.in/api_doc，在页面右侧填写申请表之后，将得到AppKey。申请界面如图7-12所示。

图7-12　PM25接口申请

我们使用的接口是获取一个城市所有监测点的AQI数据（含详情），接口地址如下:

```
http://www.pm25.in/api/querys/aqi_details.json
```

该接口的相关参数如表7-8所示。

表7-8　PM25接口参数说明

参数	含义
city	城市名称，必选参数
avg	是否返回一个城市所有监测点数据均值的标识，可选参数，默认是 true，不需要均值时传这个参数并设置为 false
stations	是否只返回一个城市均值的标识，可选参数，默认是 yes，不需要监测点信息时传这个参数并设置为 no

该接口调用举例如下（以深圳为例）：

```
http://www.pm25.in/api/querys/aqi_details.json?city=%E6%B7%B1%E5%9C%B3&avg=true&stations=no&token=GmBqBNJhqwsNf19Fwqdy
```

返回结果列表如下所示：


```
[
    {
        "aqi": 71,
        "area": "深圳",
        "co": 1.365,
        "co_24h": 1.156,
        "no2": 51,
        "no2_24h": 35,
        "o3": 34,
        "o3_24h": 68,
        "o3_8h": 35,
        "o3_8h_24h": 61,
        "pm10": 85,
        "pm10_24h": 69,
        "pm2_5": 51,
        "pm2_5_24h": 48,
        "quality": "良",
        "so2": 7,
        "so2_24h": 7,
        "primary_pollutant": "颗粒物(PM2.5)",
        "time_point": "2014-02-17T10:00:00Z"
    }
]
```


上述返回结果字段说明如表7-9所示。

表7-9　PM25接口返回字段说明

字段	字段说明
aqi	空气质量指数 (AQI)，即 air quality index，是定量描述空气质量状况的无纲量指数
area	城市名称
position_name	监测点名称
station_code	监测点编码
co	一氧化碳 1 小时平均
co_24h	一氧化碳 24 小时滑动平均
no2	二氧化氮 1 小时平均
no2_24h	二氧化氮 24 小时滑动平均
o3	臭氧 1 小时平均
o3_24h	臭氧日最大 1 小时平均
o3_8h	臭氧 8 小时滑动平均
o3_8h_24h	臭氧日最大 8 小时滑动平均
pm10	颗粒物（粒径小于等于 10μm）1 小时平均
pm10_24h	颗粒物（粒径小于等于 10μm）24 小时滑动平均

（续）

字段	字段说明
pm2_5	颗粒物（粒径小于等于 2.5μm）1 小时平均
pm2_5_24h	颗粒物（粒径小于等于 2.5μm）24 小时滑动平均
so2	二氧化硫 1 小时平均
so2_24h	二氧化硫 24 小时滑动平均
primary_pollutant	首要污染物
quality	空气质量指数类别，有“优、良、轻度污染、中度污染、重度污染、严重污染”6 类
time_point	数据发布的时间

7.4.2 开发实现

根据API接口，我们编写空气质量查询的代码如下：

```
 1 function getAirQualityChina($city)
 2 {
 3     $url =
"http://www.pm25.in/api/querys/aqi_details.json?avg=true&stations=no&city=".urlencode($city)."&token=5j1znBVAsnSf5xQyNQyq";
 4
 5     $ch = curl_init();
 6     curl_setopt($ch, CURLOPT_URL, $url);
 7     curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
 8     $output = curl_exec($ch);
 9     curl_close($ch);
10
11     $cityAir = json_decode($output, true);
12     if (isset($cityAir['error'])){
13         return $cityAir['error'];
14     }else{
15         $result = "空气质量指数(AQI)：".$cityAir[0]['aqi']."\n".
16                   "空气质量等级：".$cityAir[0]['quality']."\n".
17                   "细颗粒物(PM2.5)：".$cityAir[0]['pm2_5']."\n".
18                   "可吸入颗粒物(PM10)：".$cityAir[0]['pm10']."\n".
19                   "一氧化碳(CO)：".$cityAir[0]['co']."\n".
20                   "二氧化氮(NO2)：".$cityAir[0]['no2']."\n".
21                   "二氧化硫(SO2)：".$cityAir[0]['so2']."\n".
22                   "臭氧(O3)：".$cityAir[0]['o3']."\n".
23                   "更新时间：".preg_replace("/([a-zA-Z])/i", " ", $cityAir[0]['time_point']);
24         $aqiArray = array();
25         $aqiArray[] = array("Title" =>$cityAir[0]['area']."空气质量", "Description" =>
$result, "PicUrl" =>"", "Url" =>"");
26         return $aqiArray;
27     }
28 }
```

上述代码解读如下：

第3行：构造查询请求URL。

第5行~第9行：使用cURL获取查询结果。

第12行：进行查询结果异常判断。

第15行~第23行：拼接结果，组成字符串。

第24行~第25行：构造图文消息。

一个查询成功的结果如图7-13所示。

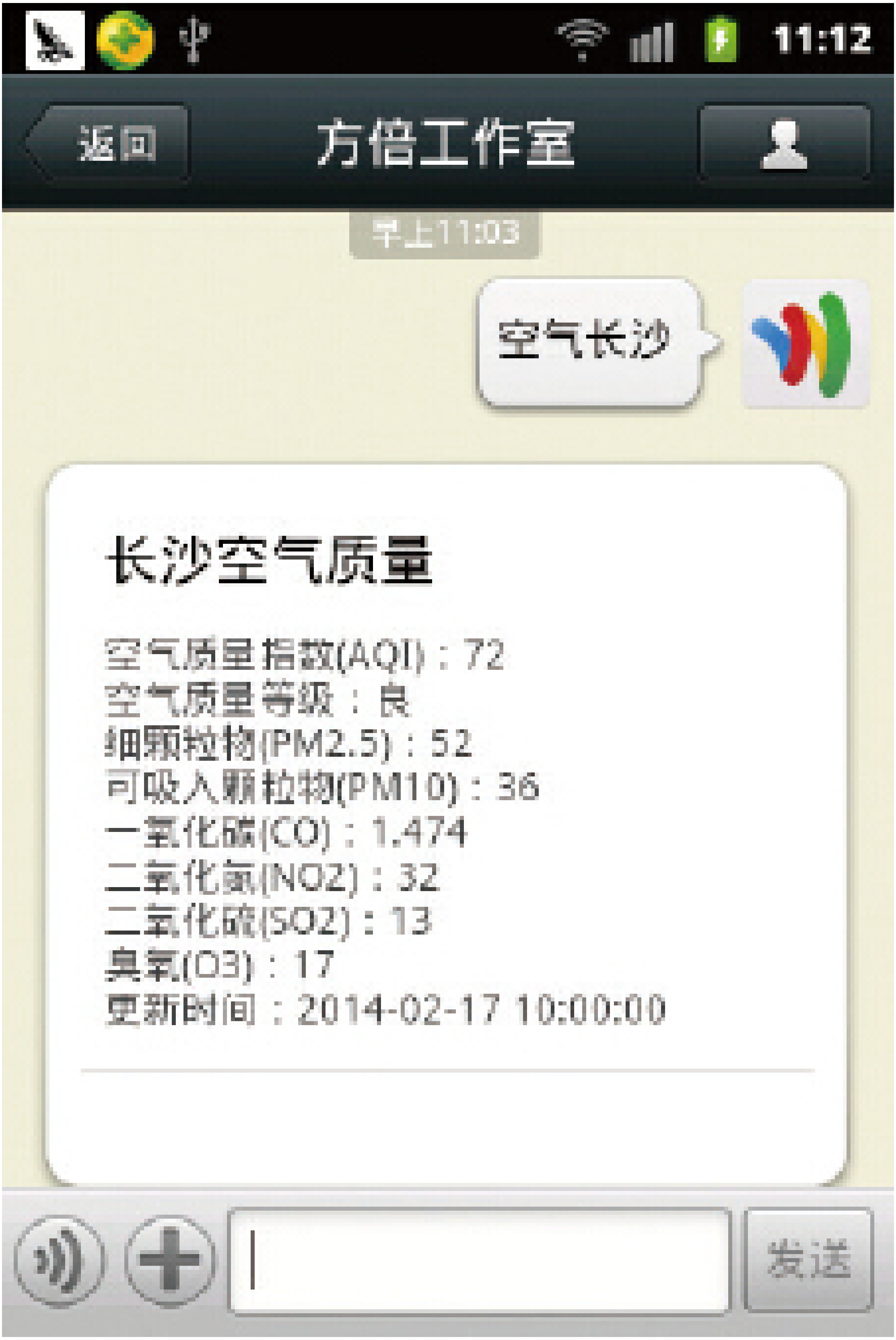

11:12
返回
方倍工作室
早上11:03
空气长沙
长沙空气质量
空气质量指数(AQI)：72
空气质量等级：良
细颗粒物(PM2.5)：52
可吸入颗粒物(PM10)：36
一氧化碳(CO)：1.474
二氧化氮(NO2)：32
二氧化硫(SO2)：13
臭氧(O3)：17
更新时间：2014-02-17 10:00:00
发送

图7-13 空气质量查询

7.5 股票行情及分析

股票是国家经济的晴雨表。股市发展与整个国民经济发展息息相关。随着炒股的人数日益增多，人们对股市行情信息获取的需求也相应增加。本节介绍在微信上开发股票行情以及个股分析查询功能。

7.5.1 行情数据

目前提供股票行情数据接口的网站主要有国内的新浪和国外的雅虎。其中雅虎能提供全球主要证券交易所的数据，包括中国。新浪也提供沪深两市几乎所有板块的股票行情，包括上海A股、上海B股、深圳A股、深圳B股、中小板、创业板等。

这里使用新浪的股票数据接口。新浪的查询接口举例如下（以中兴通讯的股票代码000063为例）：

```
http://hq.sinajs.cn/list=sz000063
```

list字段填写为股票代码或者股票代码列表（以逗号分开），股票代码前面需要添加sz或sh，表示该股票上市所在地是在深圳证券交易所或上海证券交易所。

返回结果如下所示：

```
var hq_str_sz000063="中兴通讯,14.35,14.35,14.23,14.38,14.04,14.23,14.24,45365170,644189184.63,300931,14.23,176812,14.22,32800,14.21,62800,14.20,13100,14.19,131471,14.24,186298,1
```

返回结果是个JavaScript变量，变量名称hq_str_sz000063，即字符“hq_str_”加上list字段值。而该变量的值则是需要返回的行情数据，数据用逗号隔开，一共33项，各项字段说明如表7-10所示。

表7-10 新浪股票接口返回字段说明

值	含义	值	含义
中兴通讯	股票名字	14.38	今日最高价
14.35	今日开盘价	14.04	今日最低价
14.35	昨日收盘价	14.23	竞买价，即买一报价
14.23	当前价格	14.24	竞卖价，即卖一报价

（续）

值	含义	值	含义
45365170	成交的股票数，一百股为一手	14.24	卖一报价
644189184.63	成交金额，单位元	186298	卖二股数
300931	买一股数	14.25	卖二报价
14.23	买一报价	95899	卖三股数
176812	买二股数	14.26	卖三报价
14.22	买二报价	66800	卖四股数
32800	买三股数	14.27	卖四报价
14.21	买三报价	69200	卖五股数
62800	买四股数	14.28	卖五报价
14.20	买四报价	2014-02-18	更新日期
13100	买五股数	15:37:00	更新时间
14.19	买五报价	00	不详
131471	卖一股数		

在新浪财经行情中心的找到中兴通讯000063，该股票的页面地址如下：

```
http://finance.sina.com.cn/realstock/company/sz000063/nc.shtml
```

其行情数据如图7-14所示。

图7-14 中兴通讯行情

对比图7-14与接口中的数据可以得知，数据是一致的。

7.5.2 个股分析

随着技术的不断提升，证券公司的股票系统更加实用、智能，并能从多方面对股票进行分析，使股民朋友获得更快、更全、更好的服务，以实现最大程度的赢利。

国海证券提供手机版个股诊断页面，内容包括基本面分析、趋势分析，以及分析师评级。

其分析页面地址为（以中兴通讯的股票代码000063为例）：

```
http://m.ghzq.cn/weixin/index.aspx?code=000063
```

页面在微信中打开显示如图7-15所示。

我们将把页面中有价值的内容抓取出来，然后在微信中转成文本或图文消息显示。

图7-15 个股诊断

7.5.3　开发实现

股票行情查询的核心代码如下：

```
 1 function getStockInfo($stockcode)
 2 {
 3     if (!preg_match("/^\d{6}$/",$stockcode)){
 4         return "发送股票加上6位数字代码，例如"股票000063"";
 5     }
 6     $stockIndex = array(
 7       '999999' => 'sh000001',
 8       '399001' => 'sz399001',
 9       '000300' => 'sh000300',
10       '399005' => 'sz399005',
11       '399006' => 'sz399006',
12       '000003' => 'sh000003'
13     );
14     if(array_key_exists($stockcode, $stockIndex)){
15         $url = "http://hq.sinajs.cn/list=".$stockIndex[$stockcode];
16     }else {
17         $exchange = (substr($stockcode,0,1) < 5)?"sz":"sh";
18         $url = "http://hq.sinajs.cn/list=".$exchange.$stockcode;
19     }
20     $ch = curl_init();
21     curl_setopt($ch, CURLOPT_URL, $url);
22     curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
23     $data = curl_exec($ch);
24     $result = iconv("GBK", "UTF-8//IGNORE", $data);
25
26     $start = strpos($result,'"');   //第一次出现的位置
27     $last  = strripos($result,'"'); //最后一次出现的位置
28     $stockStr = substr($result, $start + 1, $last - $start - 1);
29     $stockArray = explode(",",$stockStr);
30
31     if (count($stockArray) <> 33){ return "不存在的股票代码？"; }
32
33     $stockTitle = $stockArray[0]."[".$stockcode."]";
34     $stockInfo = "最新：".$stockArray[3]."\n".
35                  "涨跌：".round($stockArray[3]-$stockArray[2], 3)."\n".
36                  "涨幅：".round(($stockArray[3]-$stockArray[2])/ $stockArray[2]*100, 3)."%%\n".
37                  "今开：".$stockArray[1]."\n".
38                  "昨收：".$stockArray[2]."\n".
39                  "最高：".$stockArray[4]."\n".
40                  "最低：".$stockArray[5]."\n".
41                  "总手：".
42                     ((substr($stockcode,0,1) != 3)?
43                         (array_key_exists($stockcode, $stockIndex)? round(($stockArray[8]/100000000),3)."亿":round(($stockArray[8]/1000000),3)."万")
44                         :(array_key_exists($stockcode, $stockIndex)?round(($stockArray[8]/10000000000),3)."亿":round(($stockArray[8]/1000000),3)."万"))
45                     ."\n".
46                  "金额：".(array_key_exists($stockcode, $stockIndex)?round(($stockArray[9]/100000000),3)."亿":round(($stockArray[9]/10000),3)."万")."\n".
47                  "更新：".$stockArray[30]." ".$stockArray[31];
48
49     $resultArray = array();
50     $resultArray[] = array(
51                 "Title" =>$stockTitle,
52                 "Description" =>$stockInfo,
53                 "PicUrl" =>"",
54                 "Url" =>"");
55
56     return $resultArray;
57 }
```

上述代码简要说明如下：

第3行~第5行：使用正则表达式判断股票代码是否为6位数字。

第6行~第13行：将发过来的股票指数与对应的查询代码整理成数组，以便转换。

第14行~第19行：将股票指数代码直接转为相应的查询代码，而对个股的代码，自动在其前面添加相应的上市所在地标识。

第20行~第23行：使用cURL获取接口数据。

第24行：将数据编码由GBK转换成UTF-8，并且忽略错误。

第26行~第31行：截取两个双引号中间的数据，并且将其分割成数组，判断数组中单元个数是否为33，如果不是，则可能输入了不存在的股票代码。

第33行~第47行：将数组中的数据进行一定的数学处理，并且拼接成字符串。

第49行~第56行：构造一个单图文消息并返回。

股票行情查询结果如图7-16所示。

图7-16　股票行情数据

再来看下如何获取个股分析数据，使用浏览器打开上面提到的中兴通讯000063的个股诊断地址，然后在页面中单击右键，选择查看源代码（View page source），页面源码内容如下：

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
  <head>
    <title>国海证券-个股诊断</title>
    <meta charset="utf-8">
     <meta content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0;" name="viewport" />
     <meta content="black" name="apple-mobile-web-app-status-bar-style" />
    <meta content="telephone=no" name="format-detection" />
    <meta content="email=no" name="format-detection" />
     <link type="text/css" charset="UTF-8" rel="Stylesheet" href="Styles/main.css" />
  </head>
  <body>
    <div id="main">
      <header id="menu_all"></header>
      <div class="row first">
        <div>
          中兴通讯（000063）
        </div>
        <div>
          <div>
            市场喜欢程度:<div class="star"></div><div class="star"></div><div class="star"></div><div class="star"></div>
          </div>
          <div>
            <div class="F10">
              <a href="http://wap.ghzq.cn/f10/stock/f10-gh.do?tradingcode= 000063">F10</a>
            </div>
          </div>
        </div>
      </div> <div class="row">
        <div class="header">
          <div class="first"><span>|</span><span>基本面</span></div>
          <div class="last"> </div>
        </div>
        <div class="content">
          <div class="font">该公司最近一年主营收入增长率为-12.34%，净资产收益率为-2.72%；最新季报主营收入增长率为-10.00%，净资产收益率为2.55%；分析师预期未来12个月每股盈利增长率为61.71%。该
          <div class="scd-font"><span class="title">盈利增长：</span><div class="star"></div><div class="star"></div></div>
          <div class="thd-font"><span class="title">表现最佳：</span><a href='index.aspx?code=600775' style='font-size:0.85em;'>南京熊猫</a></div>
          <div class="thd-font"><span class="title">PE估值：</span><div class="star"></div><div class="star"></div></div>
          <div class="thd-font"><span class="title">表现最佳：</span><a href='index.aspx?code=000892' style='font-size:0.85em;'>星美联合</a></div>
          <div></div>
        </div>
      </div><div class="row">
        <div class="header">
          <div class="first"><span>|</span><span>技术面</span></div>
          <div class="last"> </div>
        </div>
        <div class="content">
          <div class="font">该股长期趋势相对一般，中期趋势相对较弱，短期趋势相对较弱，整体股价表现居市场中游。</div>
             <div class="scd-font"><span class="title">市场表现：</span><div class="star"></div><div class="star"></div><div class="star"></div></div>
             <div class="thd-font"><span class="title">表现最佳：</span><a href='index.aspx?code=300028' style='font-size:0.85em;'>金亚科技</a></div>
             <div class="thd-font"><span class="title">资金流入：</span><div class="star"></div><div class="star"></div></div>
          <div class="thd-font"><span class="title">表现最佳：</span><a href='index.aspx?code=300028' style='font-size:0.85em;'>金亚科技</a></div>
        </div>
      </div><div class="row">
        <div class="header">
          <div class="first"><span>|</span><span>机构认同</span></div>
          <div class="last"> </div>
        </div>
        <div class="content">
          <div class="font">当前有22位分析师对该股进行了评级，其中"强力买入"评级13人、"买入"评级9人；当前该股的综合投资评级为"买入"，机构持股比例为26.39%。该股整体机构面在市场中排名相对靠前。
```

```
61              <div class="scd-font"><span class="title">分析师情绪：</span><div class="star"></div><div class="star"></div><div class="star"></div></div>
62              <div class="thd-font"><span class="title">表现最佳：</span><a href='index.aspx?code=000063' style='font-size:0.85em;'>中兴通讯</a></div>
63              <div class="thd-font"><span class="title">机构持股：</span><div class="star"></div><div class="star"></div><div class="star"></div></div>
64              <div class="thd-font"><span class="title">表现最佳：</span><a href='index.aspx?code=300322' style='font-size:0.85em;'>硕贝德</a> </div>
65          </div>
66        </div>
67        <footer>
68          <div>2013 国海证券股份有限公司 版权所有 ICP备桂B2-20010011</div>
69              <div>Copyright ? 2013 Sealand Securities CO,LTD. All Rights Reserved</div>
70        </footer>
71      </div>
72    </body>
73 </html>
```

在上面的页面中，需要取出的数据有股票名称、基本面分析内容、技术面分析内容、机构认同内容。

第15行~第19行，有一个类名为“row first”的div区块，股票名称位于该区块下级div中的第一个。

第35、48、60行，分别是三类分析内容所在区块，区块类名为“font”，并且在该页面中“font”类不再包括其他内容。

我们使用simple_html_dom.php来抓取所需要的内容。

PHP Simple HTML DOM Parser是PHP下目前最优秀的HTML元素解析器，作者是中国台湾的S.C.Chen（陈世钦），该解析器项目主页地址为：http://simplehtmldom.sourceforge.net/。

使用simple_html_dom.php来获取个股诊断的代码如下所示：

```
1 function getStockAnalysis($stockcode)
2 {
3     if (!preg_match("/^\d{6}$/",$stockcode)){
4         return "发送分析加上6位数字代码，例如"分析000063"";
5     }
6 
7     include_once('simple_html_dom.php');
8     try {
9         $url = "http://m.ghzq.cn/weixin/index.aspx?code=".$stockcode;
10        $html_analysis = file_get_html($url);
11        if (!isset($html_analysis)){
12            $html_analysis->clear();
13        }else{
14 
15            $stock = $html_analysis->find('div[class="row first"] div', 0)->plaintext;
16             $resultArray[] = array("Title" =>trim($stock), "Description" =>"", "PicUrl" =>"", "Url" =>"");
17            //基本面
18            $fundamentals = $html_analysis->find('div[class="font"]', 0);
19            $resultArray[] = array("Title" =>str_replace("%", "%%", "【基本面】\n".$fundamentals->plaintext), "Description" =>"", "PicUrl" =>"", "Url" =>"");
20            //技术面
21            $technical = $html_analysis->find('div[class="font"]', 1);
22            $resultArray[] = array("Title" =>str_replace("%", "%%", "【技术面】\n".$technical->plaintext), "Description" =>"", "PicUrl" =>"", "Url" =>"");
23            //机构认同
24            $technical = $html_analysis->find('div[class="font"]', 2);
25             $resultArray[] = array("Title" =>str_replace("%", "%%", "【机构认同】\n".$technical->plaintext), "Description" =>"", "PicUrl" =>"","Url" =>"");
26            $html_analysis->clear();
27        }
28    }catch (Exception $e){
29 
30    }
31    return $resultArray;
32 }
```

上述代码简要说明如下：

第3行~第5行：使用正则表达式判断股票代码是否为6位数字。

第7行：引用该解析器的库文件。

第9行：构造要诊断的页面地址。

第10行：从页面地址中创建文档对象模型。

第11行~第13行：判断模型是否创建成功。

第15行~第16行：查找class为“row first”的div元素，并在该元素下再找div元素，符合这样条件的第一个结果即为股票标题，取其文本值放入图文消息中。

第17行~第19行：查找class为“font”的div元素，符合这样条件的第一个结果为基本面内容，取其文本值放入图文消息中。

第20行~第22行：查找class为“font”的div元素，符合这样条件的第二个结果为技术面内容，取其文本值放入图文消息中。

第23行~第25行：查找class为“font”的div元素，符合这样条件的第三个结果为机构认同内容，取其文本值放入图文消息中。

第26行：清理文档对象，释放资源。

第31行：返回图文消息。

个股分析的结果如图7-17所示。

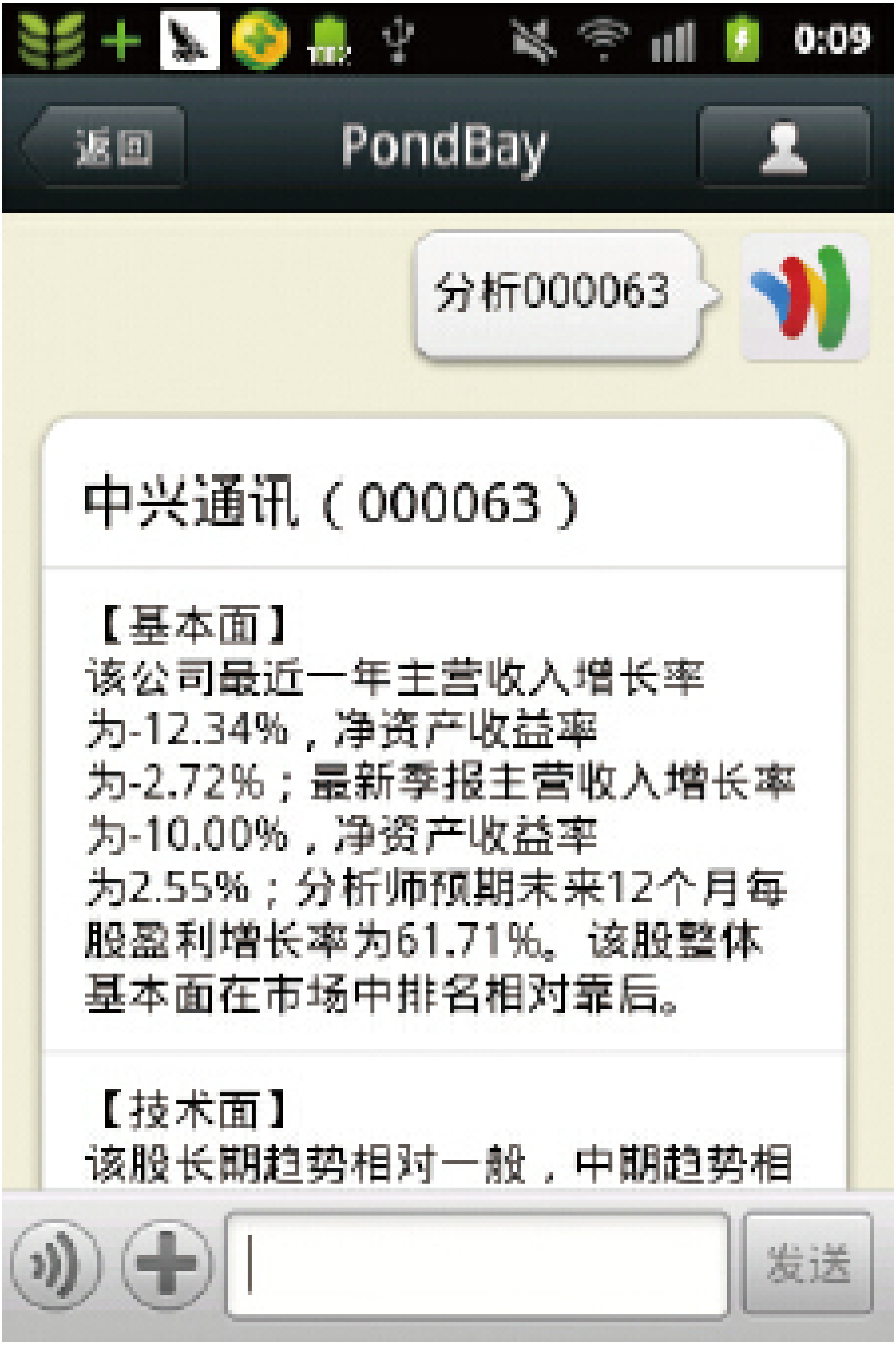

0:09
返回
PondBay
分析000063
中兴通讯（000063）
【基本面】
该公司最近一年主营收入增长率为-12.34%，净资产收益率为-2.72%；最新季报主营收入增长率为-10.00%，净资产收益率为2.55%；分析师预期未来12个月每股盈利增长率为61.71%。该股整体基本面在市场中排名相对靠后。
【技术面】
该股长期趋势相对一般，中期趋势相
发送

图7-17　个股分析结果

7.6　苹果产品信息查询

苹果公司的产品最近几年如日中天，拥有苹果产品的人也越来越多，产品的保修期以及技术支持服务器的查询数量也越来越大。本节介绍如何在微信上实现苹果产品信息查询功能。

7.6.1　过程分析

在这一节中，我们使用Fiddler这个软件来分析苹果产品信息的查询过程。Fiddler是一个免费的Web调试工具，用来监测本地计算机和Internet之间所有的HTTP通信。官方下载地址为：http://www.telerik.com/download/fiddler。

Fiddler的软件界面如图7-18所示。

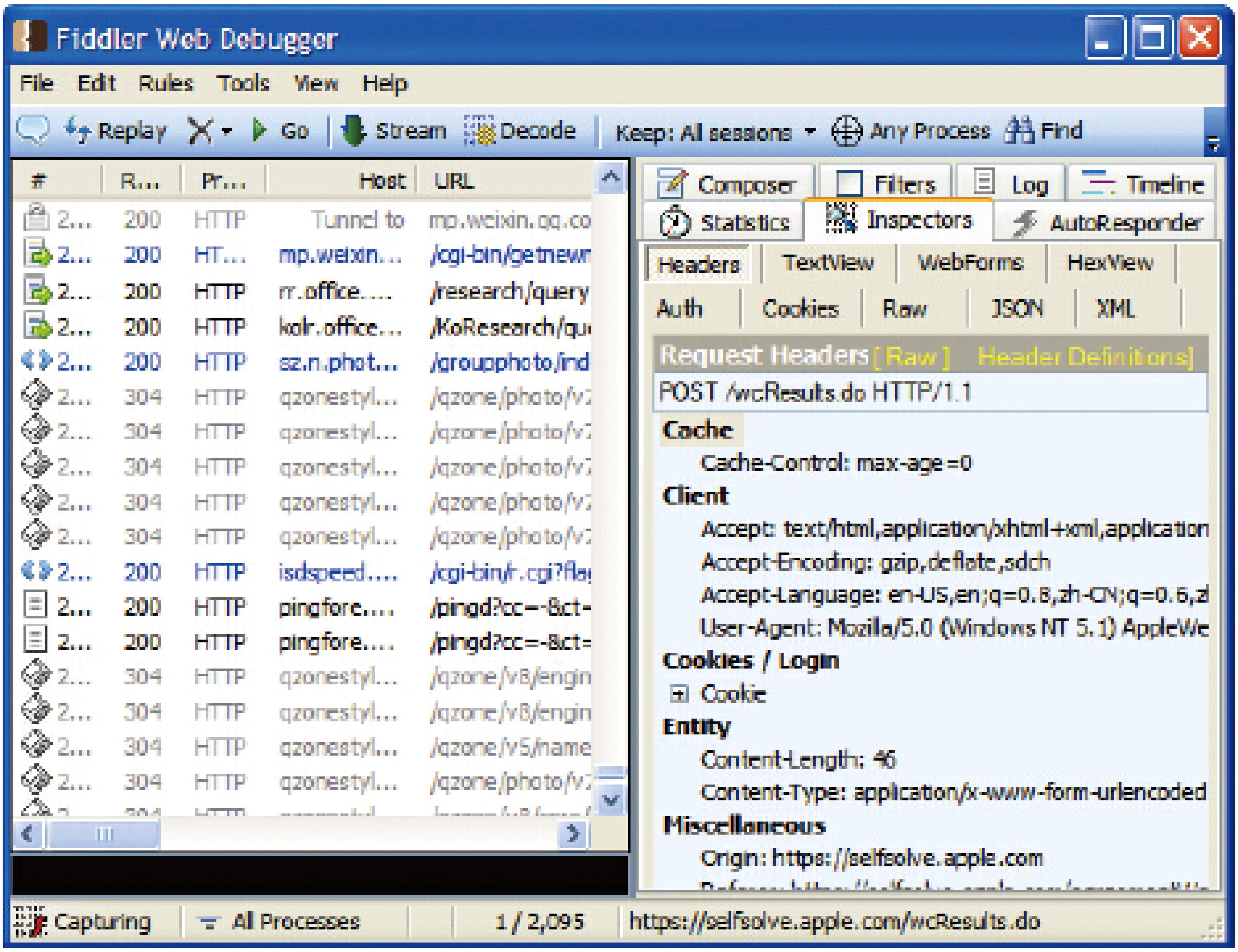


图7-18　Fiddler

启动Fiddler后，软件自动处在监听状态，然后在浏览器中打开苹果公司“技术支持-查看您的保修服务和支持期限”页面，页面地址为：

```
https://selfsolve.apple.com/agreementWarrantyDynamic.do?
```

可以看到苹果产品查询入口，如图7-19所示。

图7-19　苹果产品查询入口

在输入框中填写一个苹果产品的序列号，然后单击“继续”，返回页面如图7-20所示。

图7-20　苹果产品查询结果

在Fiddler中，可以看到已经监听到查询请求，查询请求的HTTP头部如图7-21所示。

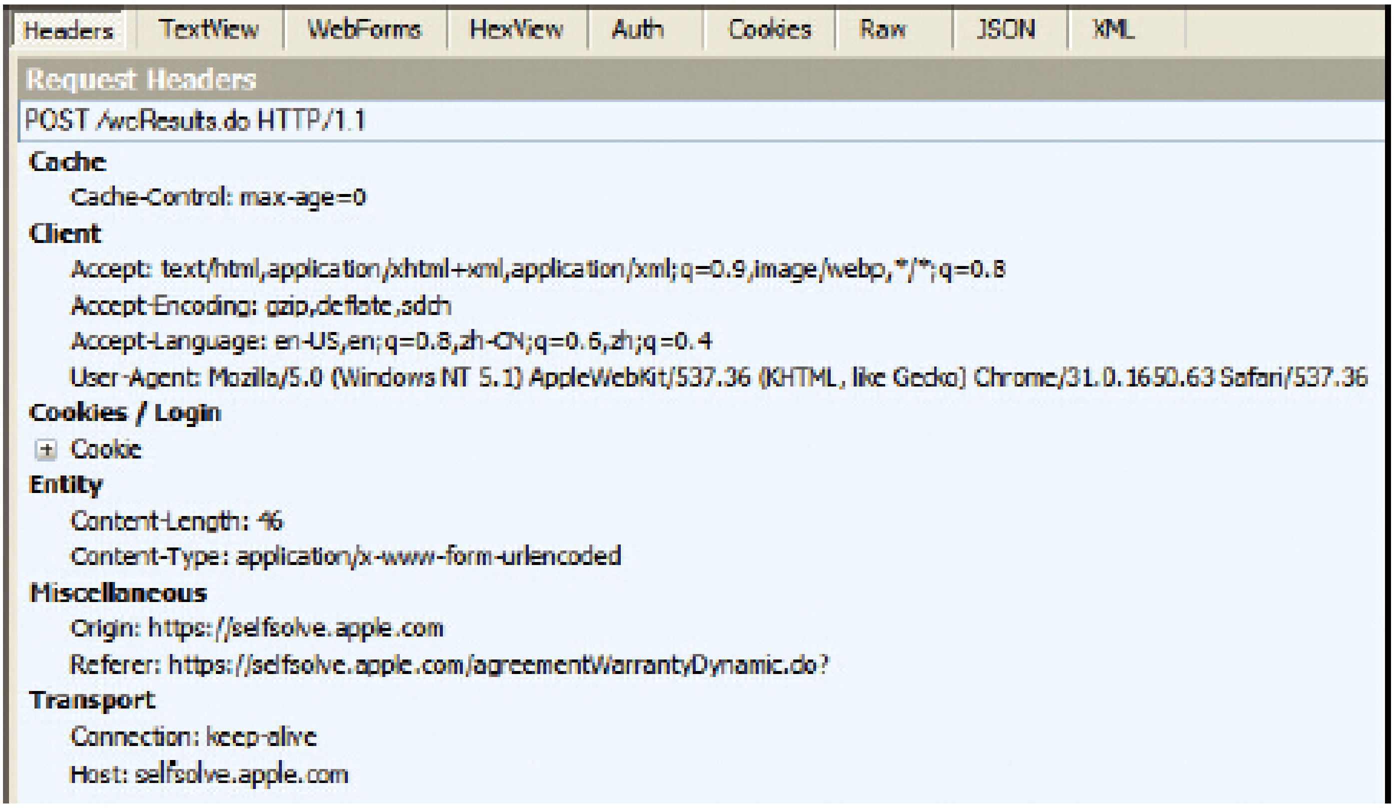


图7-21　HTTP请求头部

该请求是一个POST请求，请求的内容体如图7-22所示。

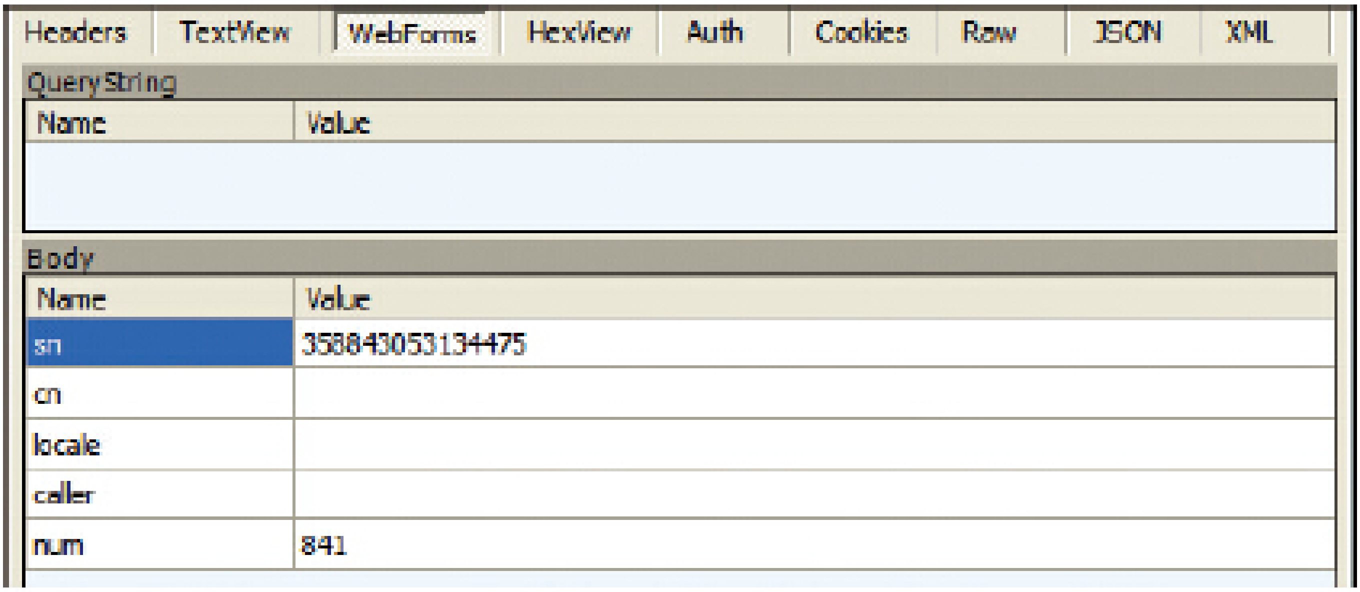


图7-22　请求Web表单

该请求的字符串形式如下：

```
sn=358843053134475&cn=&locale=&caller=&num=841
```

其中sn即为要查询的序列号，而num这一参数在经过多次尝试分析后可知它是一个随机数。

查看查询结果页面的源代码，可以找到查询结果的数据在下面这一段：

```
<script type="text/javascript">
var warrantyPage;
//instantiate warranty checker
var warrantyChecker = new WarrantyChecker();
var hasErrors = false;

Event.observe(window, 'load', function() {

warrantyPage = new WCPage();
warrantyPage.warrantycheck.initialize(230147);




    warrantyPage.warrantycheck.hidePanel();
```

```
16     $('cookieSpinner').hide();
17
18
19              warrantyPage.warrantycheck.hideRow('iphonenotactivated');
20
21                  $('results').show();
22                  warrantyPage.warrantycheck.showTitle('status');
23
24                  var regTextKeyLong = '借助已验证的购买日期，Apple 可以快速找到您的产品并为您提供所需帮助。';
25                  $('registration-text').update(regTextKeyLong);
26                  warrantyPage.setClassAndShow('registration', 'green', 'regi-
stration-true');
27
28                  warrantyPage.warrantycheck.displayProductInfo ('https://km.support.apple.com.edgekey.net/kb/securedImage.jsp?configcode= FRC6&size=72x72', 'iPhone 5S', true
29
30                  warrantyPage.warrantycheck.displayPHSupportInfo( true, '电话技术支持：有效', '在最初90天内，您的产品可享受免费的电话技术支持服务。<br/>预计到期日：16 May 2014<br/><a
31
32                  warrantyPage.warrantycheck.displayHWSupportInfo (true, '维修和服务保修情况：有效', '您的产品可享受 Apple 有限保修中的硬件维修和服务。<br/>预计到期日：14 February 2015
33
34                 warrantyPage.warrantycheck.displayEligibilityInfo ('LI', false, '', '358843053134475', '很抱歉，该产品不符合 AppleCare 协议条件。', '/AgrOfferFlow.do','CHN','APP
35
36                  warrantyPage.warrantycheck.displayDisputeInfo('<div class="sosumi">如果您认为针对您的产品所显示的信息不正确，可<a href= "https://selfsolve.apple.com/Dispute.do?t
37
38
39                //Begin:Added json Object:Espressp #14774083
40                    var jsonObj = {
41                                  'phoneSupportHasCoverage' : true,
42                                  'hwSupportHasCoverage' :    true,
43                                  'hwSupportCoverageValue' :     'LI',
44                                  'hasActiveAPP':              false,
45                                  'isAPPEligible' :            false,
46                                  'isPPIEligible' :            false,
47                                  'productType' :              'iPhone',
48                                  'numDaysSinceDOP':           '4',
49                                  'hasTechToolDownload':        false
50                     };
51
52                  warrantyChecker.trackOmnitureResults(jsonObj);
53
54                  //End:Added json Object:Espressp #14774083
55
56
57 });
58
59 </script>
```

我们需要获取的有价值的信息如下：

第26行：显示产品是否为有效的购买日期。

第28行：显示产品的名称以及产品图片地址。

第30行：显示产品电话技术是否有效以及到期日期。

第32行：显示产品维修和服务保修情况是否有效以及到期日期。

我们将使用正则表达式将我们需要的信息提取出来。

7.6.2 开发实现

苹果产品信息查询代码如下：

```
function getAppleInfo($sn)
{
    $rnd = rand(100,999999);
    $post = "sn=$sn&cn=&locale=&caller=&num=$rnd";
    $url = "https://selfsolve.apple.com/wcResults.do";

    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL, $url );
    curl_setopt($ch, CURLOPT_HEADER, false);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER,1);
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
    $result=curl_exec($ch);
    curl_close($ch);

    $result = str_replace("'","",$result);
    preg_match('#setClassAndShow\((.*?)\)#is',      $result, $RegistrationInfo);//注册信息
    preg_match('#displayProductInfo\((.*?)\)#is',     $result, $ProductInfo);     //产品信息
    preg_match('#displayPHSupportInfo\((.*?)\)#is',   $result, $PHSupportInfo);   //电话支持
    preg_match('#displayHWSupportInfo\((.*?)\)#is',   $result, $HWSupportInfo);   //硬件保修

    if (empty($RegistrationInfo)){
        return "很抱歉，此序列号无效。请检查您的信息，然后再试。";
    }

    $registration = explode(",",$RegistrationInfo[1]);
    $product = explode(",",$ProductInfo[1]);
    $phsupport = explode(",",$PHSupportInfo[1]);
    $hwsupport = explode(",",$HWSupportInfo[1]);

    $phsupport_date = "";   //提取电话技术支持日期
    if (trim($phsupport['0']) == "true"){
        preg_match('#Estimated Expiration Date:(.*?)<br/>#is', $phsupport['2'].$phsupport['3'], $phsupport_date);
    }
    $hwsupport_date = "";   //提取保修服务日期
    if (trim($hwsupport['0']) == "true"){
        preg_match('#Estimated Expiration Date:(.*?)<br/>#is', $hwsupport['2'].$hwsupport['3'], $hwsupport_date);
    }

    $title = "苹果产品信息查询";
    $description = "设备名称：".trim($product['1'])."\n".
  ((preg_match("/^\d{8,20}$/",$sn))?"IMEI号":"序列号").": ".$product['3']."\n".
    "购买日期：".((trim($registration['2']) == "registration-true")?"已验证":"无效")."\n".
     "电话支持：".((trim($phsupport['0']) == "true")?"有效[".trim ($phsupport_date['1'])."]":"已过期")."\n".
     "保修服务：".((trim($hwsupport['0']) == "true")?"有效[".trim ($hwsupport_date['1'])."]":"已过期")."\n".
    "\n数据来自苹果公司官方网站";
    $picurl = $product['0'];

    $months = array(
        "01-"=>"January ", "02-"=>"February ", "03-"=>"March ",
        "04-"=>"April ", "05-"=>"May ", "06-"=>"June ",
        "07-"=>"July ", "08-"=>"August ", "09-"=>"September ",
        "10-"=>"October ", "11-"=>"November ", "12-"=>"December ",
    );
    foreach($months as $key=>$value){
        $description = str_ireplace($value, $key, $description);
    }

    $result = array();
```

```
61     $result[] =  array(
62         "Title"=>$title,
63         "Description" =>trim($description),
64         "PicUrl" =>$picurl,
65         "Url" =>""
66     );
67     return $result;
68 }
```

上述代码简要解释说明如下：

第3行：生成随机数。

第4行：各参数链接成要POST的数据。

第5行：定义要提交的URL接口，注意此接口并非在浏览器中打开的地址，而是Fiddler截获的数据提交真实地址。

第7行~第15行：使用cURL提交POST请求。由于接口使用了HTTPS协议，cURL发起请求的时候，将会检查CA证书，于是第11、12行设置为不验证证书来源及主机。第13行设置要发送的POST数组。

第17行：将 “'” 替换成空。即删除它，以便后续代码处理字符串更方便。

第18行~第21行：使用正则表达式提取注册信息、产品信息、电话支持信息、硬件保修信息的相关文字。

第23行~第25行：判断注册信息是否为空，如果为空，则输入了无效的序列号或IMEI号。

第27行~第30行：将注册信息、产品信息、电话支持信息、硬件保修信息中的字符串分割成数组。

第32行~第35行：使用正则表达式从电话支持信息数组中提取电话支持截止日期。

第36行~第39行：使用正则表达式从硬件保修信息数组中提取硬件保修截止日期。

第41行~第48行：图文消息的标题自己定义，描述信息的内容为获取到的信息连接后形成的字符串，图片URL直接从产品信息数组中取得。

第50行~第58行：将描述信息中的英文月份转换为数字月份。

第60行~第66行：构造图文消息。

第67行：返回图文消息。

在微信上查询苹果产品信息如图7-23所示。

图7-23　苹果产品信息查询

7.7　历史上的今天

唐太宗李世民说过：“以铜为镜，可以正衣冠；以古为镜，可以知兴替；以人为镜，可以明得失”。历史是生活的一面镜子，查看历史上发生的大事，可以陶冶人的情操，提高人的修养，增长人的智慧。本节介绍如果在微信公众平台上开发实现历史上的今天这一功能。

7.7.1　数据来源

在百度中搜索“历史上的今天”，返回的第一条网址就是http://www.todayonhistory.com/，并且被百度标记为“官网”。单击进入该网站，发现其收录的数据比较齐全，页面也比较规范，如图7-24所示。我们就从这个网站来获取数据。

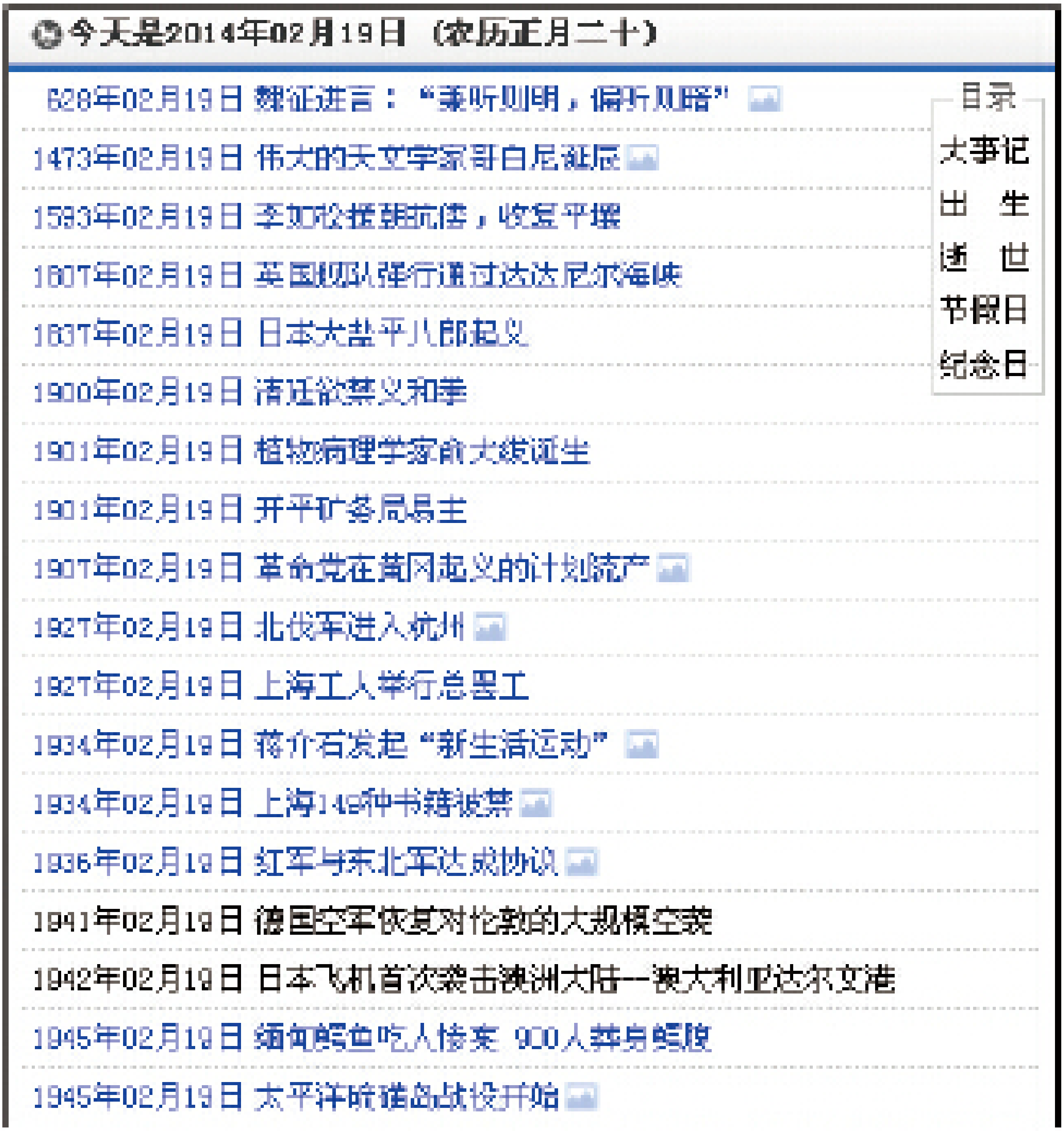


图7-24　历史上的今天

同样地，我们使用simple_html_dom来抓取所需要的页面。

查看网页源代码，事件内容部分代码显示如下：

```
<div class="main">
    <ul class="list clearfix">
      <li class="gong"><a href="/2/19/WeiZhengJinYan-JianTingZeMing-Pian-
TingZeAn.html" class="screenshot" rel="http://img.todayonhistory.com/201007/16/A114948368.jpg" title="魏征进言：“兼听则明，偏听则暗”" target="_blank" _orighref="/2/19/WeiZhengJinYa
```

```
        <li class="gong st4"><a href="/2/19/WeiDaDeTianWenXueJiaGeBaiNiDanChen.html" class="screenshot" rel="http://img.todayonhistory.com/200402/19/B2225552453.jpg" title="伟大
         <li class="gong"><a href="/2/19/LiRuSongYuanChaoKangWo-ShouFuPingRang.
html" title="李如松援朝抗倭，收复平壤" target="_blank"><em>1593年02月19日</em> <i>李如松援朝抗倭，收复平壤</i></a></li>
```

可以看到，事件内容都放置在div区块下的ul标签下的li标签中。并且按照年份时间顺序，由远及近排列。我们可以通过循环遍历的方式提取相应数据。

7.7.2 开发实现

历史上的今天的查询代码如下：

```
1 function getHistoryInfo()
2 {
3     include_once('simple_html_dom.php');
4     try {
5         $url = "http://www.todayonhistory.com/";
6         $html_analysis = file_get_html($url);
7         if (!isset($html_analysis)){
8             $html_analysis->clear();
9             return "获取失败，请联系方倍工作室";
10        }else{
11            $contentStr = "历史上的":date("m")."月".date("d")."日:\n";
12            foreach($html_analysis->find('div[class="main"] ul[class= "list clearfix"] li') as $item) {
13                $contentStr .= str_replace(date("m")."月".date("d"). "日","",$item->plaintext)."\n";
14                if (strlen($contentStr) > 2000){break;}
15            }
16            $html_analysis->clear();
17            return trim($contentStr);
18        }
19    }catch (Exception $e){
20    }
21 }
```

上述代码简要解释说明如下：

第3行：引用该解析器的库文件。

第5行：定义要访问的网址URL。

第6行：从页面地址中创建文档对象模型。

第7行~第9行：判断模型是否创建成功。

第11行：定义内容最前面显示的引导话。

第12行~第15行：查找class为“main”的div元素，并在该元素下查找class为“list clearfix”的ul元素，再在下面找li元素，符合这样条件的均为事件内容，通过循环遍历的方式来获取，取其纯文本内容。由于每条内容都包含“×月×日”这样的文字，在第13行将这样的文字统一移除掉。最后，由于微信能接受的一条文本消息最大长度为2048字节，在第14行做出长度限制，超过2000以后就不再获取下面的事件内容了，以防消息内容超过最大长度而被微信服务器丢弃。

第16行：清理文档对象，释放资源。

第17行：返回文本消息。

在微信上查询历史上的今天信息如图7-25所示。

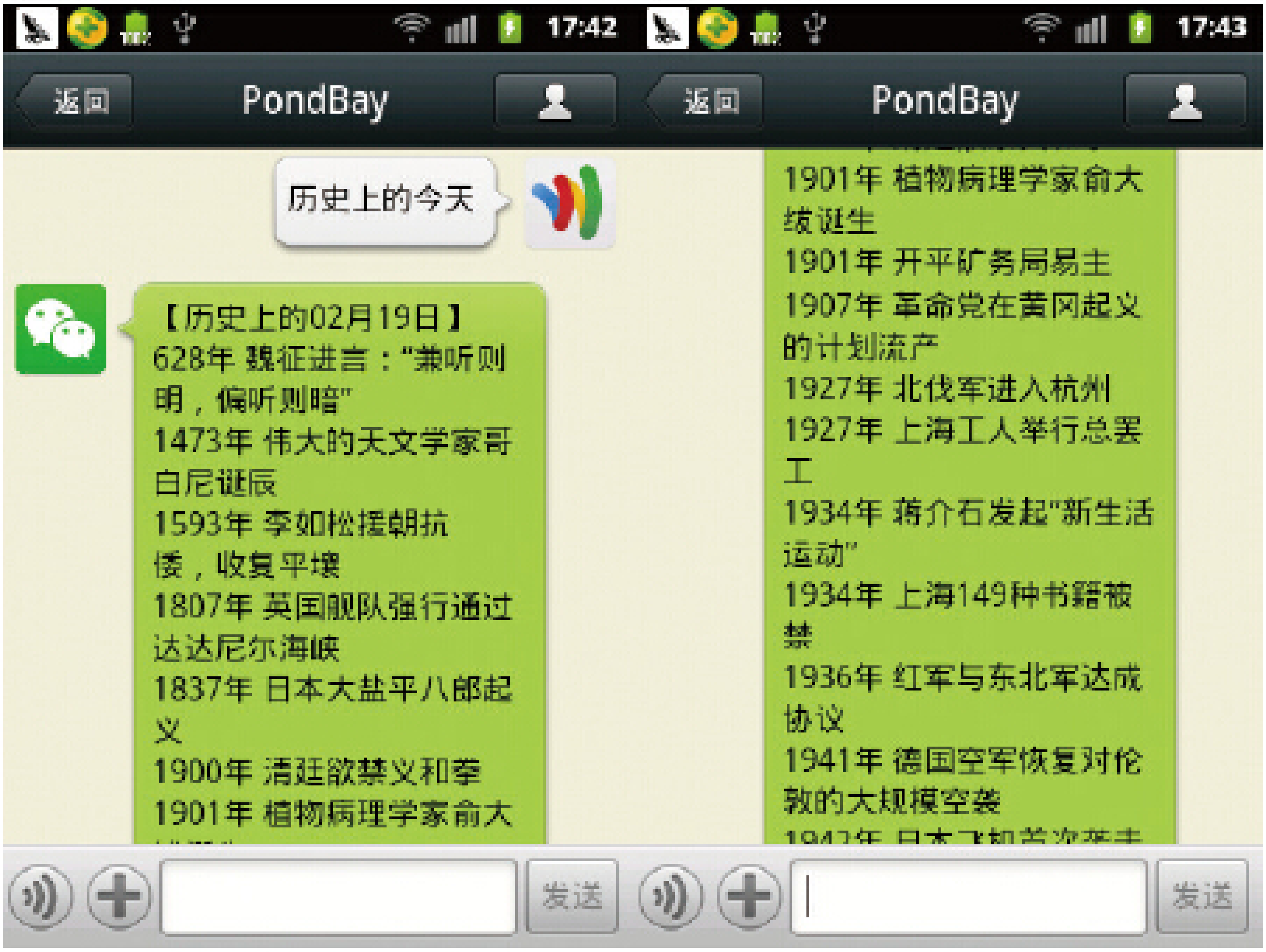


图7-25　微信查看历史上的今天

7.8　附近搜索

当我们到一个陌生的地方，想查看附近有什么银行、医院、酒店，甚至附近有哪些公交车站，经常会不知道从何找起。现在微信有了地理位置接口，我们可以基于这个接口来实现附近目标搜索的功能。

本节将介绍如何实现附近搜索功能。

7.8.1　百度地图

百度地图API是一套为开发者免费提供的基于百度地图的应用程序接口，包括JavaScript、iOS、Andriod、静态地图、Web服务等多种版本，提供基本地图、位置搜索、周边搜索、公交驾车导航、定位服务、地理编码及逆地理编码等丰富功能。通过地图API，一方面解决移动开发者服务器端日益增长的海量位置数据的存储维护压力，另一方面彻底解决所有LBS开发者基于位置数据的高并发检索瓶颈。

调用百度地图API需要注册百度账号，然后再申请ak（访问密钥），申请地址为

```
http://lbsyun.baidu.com/apiconsole/key?application=key
```

进入申请地址页面后，如图7-26所示。

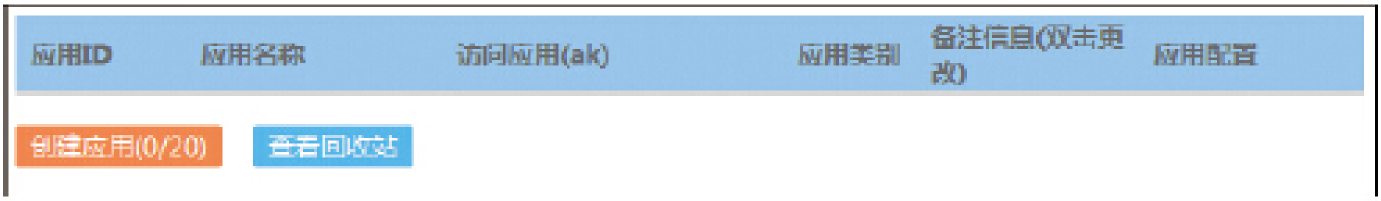


图7-26　应用列表

单击“创建应用”按钮，弹出创建应用窗口，填写相应内容，如图7-27所示。

图7-27　创建应用

单击“确认”按钮后，回到控制台，在页面上可以看到，已经成功申请到ak，值为：MgBALVVeCd8THVBi6gPdvsvG，如图7-28所示。

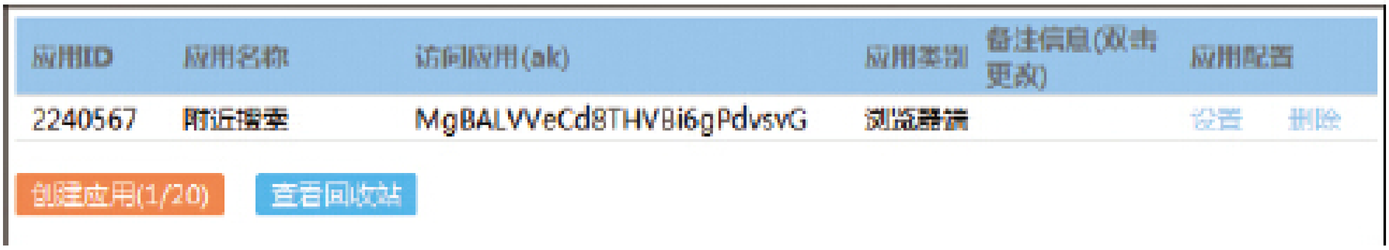


图7-28　申请到ak

百度地图Place API是一类简单的HTTP接口，用于返回查询某个区域的某类POI数据，且提供单个POI的详情查询服务，用户可以使用C#、C++、Java等开发语言发送HTTP请求且接收JSON、XML的数据。

Place API提供区域检索POI服务、POI详情服务与团购信息检索服务、商家团购详情查询。区域检索POI服务提供三种区域检索方法：城市内检索、矩形检索、圆形区域检索。

我们使用圆形区域检索来实现附近搜索功能。

place区域检索POI服务接口如下：

```
http://api.map.baidu.com/place/v2/search
```

该接口的参数说明如表7-11所示。

表7-11　Place API参数说明

参数	是否必须	默认值	格式举例	含义
location	是	无	38.76623,116.43213 lat< 纬度 >,lng< 经度 >	周边检索中心点，不支持多个点
radius(r)	否	无	2000	周边检索半径，单位为米
q(query)	是	无	中关村、ATM、百度大厦	检索关键字，周边检索和矩形区域内检索支持多个关键字并集检索，不同关键字间以 $ 符号分隔，最多支持 10 个关键字检索。如 :"银行 $ 酒店"
tag	否	无	日式烧烤 / 铁板烧、朝外大街	标签项，与 q 组合进行检索
output	否	xml	json 或 xml	输出格式为 json 或者 xml
scope	是	1	1、2	检索结果详细程度。取值为 1 或空，则返回基本信息；取值为 2，返回检索 POI 详细信息
filter	否	无	filter=industry_type:cater |sort_name:price |sort_rule:0 |price_section:100,200 |groupon:0 |discount:0	检索过滤条件，当 scope 取值为 2 时，可以设置 filter 进行排序。 industry_type：行业类型 sort_name：排序字段 sort_rule：排序规则，取值如下：0：从高到低，1：从低到高； price_section：价格区间； groupon：是否有团购，1 为有团购，0 为无团购； discount：是否打折，1 为有打折，0 为无打折
page_size	否	10	10	范围记录数量，默认为 10 条记录，最大返回 20 条。多关键字检索时，返回的记录数为关键字个数 *page_size
page_num	否	0	0、1、2	分页页码，默认为 0,0 代表第一页，1 代表第二页，以此类推
ak	是	无	E4805d16520de693a3fe707cdc962045	用户的访问密钥，必填项。v2 之前该属性为 key
sn	否	无		用户的权限签名
timestamp	否	无		设置 sn 后该值必填

该接口调用举例如下：

```
http://api.map.baidu.com/place/v2/search?ak=MgBALVVeCd8THVBi6gPdvsvG&output=json&query=%E9%93%B6%E8%A1%8C&page_size=5&page_num=0&scope=2&location=39.915,116.404&radius=2000&fil
```

返回结果列表如下所示：

```
{
    "status":0,
    "message":"ok",
    "total":142,
    "results":[
        {
            "name":"北京银行ＡＴＭ",
            "location":{
                "lat":39.914663,
                "lng":116.402225
            },
            "address":"北京市东城区东长安街",
            "street_id":"08db2caeab1dc6dcfb2213da",
            "uid":"08db2caeab1dc6dcfb2213da",
            "detail_info":{
                "distance":155
            }
        },
        {
            "name":"交通银行ＡＴＭ",
            "location":{
                "lat":39.917284,
                "lng":116.403677
            },
            "address":"北京市东城区",
            "uid":"7af671e39c0b6c1158f6be9f",
            "detail_info":{
                "distance":255
            }
        },
        {
            "name":"中国银行ＡＴＭ",
            "location":{
                "lat":39.912954,
                "lng":116.396091
            },
            "address":"北京市西城区西长安街２号",
            "street_id":"82cf55ea33c0f0eefbdc856b",
            "uid":"82cf55ea33c0f0eefbdc856b",
            "detail_info":{
                "distance":711
            }
        },
        {
            "name":"中信银行atm",
            "location":{
                "lat":39.916226,
```

```
            "lng":116.412688
        },
        "address":"南湾子胡同1号北京香江戴斯酒店",
        "street_id":"1ef6d8865a6870889de32120",
        "uid":"1ef6d8865a6870889de32120",
        "detail_info":{
            "distance":753
        }
    },
    {
        "name":"原麦加利银行",
        "location":{
            "lat":39.908533,
            "lng":116.406818
        },
        "address":"东交民巷39",
        "street_id":"6cc9b6851fa53eb715c9057e",
        "uid":"6cc9b6851fa53eb715c9057e",
        "detail_info":{
            "distance":757
        }
    }
  ]
}
```

上述返回结果字段说明如表7-12所示。

表7-12　Place API返回字段说明

名称	类型	说明
status	int	本次 API 访问状态，如果成功返回 0，如果失败返回其他数字
message	string	对 API 访问状态值的英文说明，如果成功返回 "ok"，并返回结果字段，如果失败返回错误说明
total	int	检索总数，用户请求中设置了 page_num 字段才会出现 total 字段
name	string	POI 名称
location	object	POI 经纬度坐标
lat	float	纬度值
lng	float	经度值
address	string	POI 地址信息
telephone	string	POI 电话信息
uid	string	POI 的唯一标识
detail_info	object	POI 的扩展信息，仅当 scope=2 时，显示该字段，不同的 POI 类型，显示的 detail_info 字段不同
distance	int(32)	距离中心点的距离
type	string	所属分类，如 “hotel”、“cater”
tag	string	标签
detail_url	string	poi 的详情页
price	string	poi 商户的价格
shop_hours	string	营业时间
overall_rating	string	总体评分
taste_rating	string	口味评分
service_rating	string	服务评分

（续）

名称	类型	说明
environment_rating	string	环境评分
facility_rating	string	星级（设备）评分
hygiene_rating	string	卫生评分
technology_rating	string	技术评分
image_num	string	图片数
groupon_num	int	团购数
discount_num	int	优惠数
comment_num	string	评论数
favorite_num	string	收藏数
checkin_num	string	签到数

7.8.2　开发实现

本节我们使用缓存的方式来存储用户的地理位置坐标记录。

SAE为开发者提供Memcache服务。Memcache是一个高性能、分布式内存对象缓存系统，通过在内存里维护一个统一的巨大的hash表，它能够用来存储各种格式的数据，包括图像、视频、文件以及数据库检索的结果等。简单地说就是将数据调用到内存中，然后从内存中读取，从而大大提高读取速度。

在SAE中的服务管理列表中，可以看到Memcache一项，如图7-29所示。

图7-29　SAE的Memcache

单击“点此初始化MC”按钮，弹出初始化MC窗口，根据自己的需求填写要使用的缓存容量，如图7-30所示。

单击“确定”按钮后，缓存创建成功，可以看到Memcache的基本信息，如图7-31所示。

图7-31　Memcache

至此，我们就可以开始使用Memcache的功能了。

在缓存中保存用户地理位置信息代码如下：

```
1 function setLocation($openid, $locationX, $locationY)
2 {
3     $mmc = memcache_init();
4     if($mmc == true){
5         $location = array("locationX"=>$locationX, "locationY"=> $locationY);
6         memcache_set($mmc, $openid, json_encode($location), 60);
7         return "您的位置已缓存。\n现在可发送"附近"加目标的命令，如"附近酒店"， "附近加油站"。";
8     }
9     else{
10         return "未启用缓存，请先开启服务器的缓存功能。";
11     }
12 }
```

上述代码简要解释说明如下：

第3行：初始化MC链接。

第5行：将坐标信息存在数组中。

第6行：使用memcache_set存入MC信息。Key值为openid，Value为JSON编码后的坐标数组，过期时间设置为60秒。

从缓存中读取用户地理位置信息代码如下：

```
function getLocation($openid)
{
    $mmc = memcache_init();
    if($mmc == true){
        $location = memcache_get($mmc, $openid);
        if (!empty($location)){
            return json_decode($location,true);
        }else{
            return "请先发送位置给我！\n单击底部的'+'号，再选择'位置'，等地图显示出来以后，单击'发送'";
        }
    }
    else{
        return "未启用缓存，请先开启服务器的缓存功能。";
    }
}
```

上述代码简要解释说明如下：

第3行：初始化MC链接。

第5行：使用memcache_set读取MC信息，返回以openid作为key存储的元素的值。

第7行：将取得到的JSON格式的坐标信息进行编码，转回数组。

在百度地图中搜索附近的代码如下：

```
1 function catchEntitiesFromLocation($entity, $x, $y, $radius)
2 {
3     $url = "http://api.map.baidu.com/place/v2/search?ak=MgBALVVeCd8THVBi6gPdvsvG&output=json&query=".$entity."&page_size=5&page_num=0&scope=2&location=".$x.",".$y."&radius="
4 
5     $ch = curl_init();
6     curl_setopt($ch, CURLOPT_URL, $url);
7     curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
8     $output = curl_exec($ch);
9 
10    $data = json_decode($output, true);
11 
12    if ($data['status'] != 0){
13        return $data['message'];
14    }
15 
16    $results = $data['results'];
17    if (count($results) == 0){
18        return "附近没有找到".$entity;
19    }
20    $shopArray = array();
21    $shopArray[] = array("Title"=>"附近的".$entity, "Description"=>"", "PicUrl"=>"", "Url"=>"");
22    for ($i = 0; $i < count($results); $i++) {
23        $shopArray[] = array(
24            "Title"=>"【".$results[$i]['name']."】<".$results[$i]['detail_info']['distance']."米>\n".$results[$i]['address'].
25            (isset($results[$i]['telephone'])?"\n".$results[$i]['telephone']:""),
26            "Description"=>"",
27            "PicUrl"=>"",
28     "Url"=>(isset($results[$i]['detail_info']['detail_url'])? ($results[$i]['detail_info']['detail_url']):""));
29    }
30    return $shopArray;
31 }
```

上述代码简要解释说明如下：

第1行：函数的各项参数说明，$entity：用户要搜索的内容，比如“酒店”、“银行”；$x：纬度坐标；$y：经度坐标；$radius：搜索半径，以坐标为中心点。

第3行：构造URL。

第5行~第8行：使用cURL获取查询结果。

第10行：将JSON格式的字符串解码成数组。

第12行~第14行：判断接口返回是否异常。

第16行~第19行：判断接口是否有内容返回。

第20行~第30行：拼接内容，封装成图文消息返回。

而公众账号接收到地址位置消息时处理代码如下：

```
private function receiveLocation($object)
{
    include("location.php");
    $content = setLocation($object->FromUserName,(string)$object-> Location_X, (string)$object->Location_Y);
    $result = $this->transmitText($object, $content);
    return $result;
}
```

上述代码中将微信用户的OpenID以及经纬度坐标存放在缓存中。

公众账号收到用户查询请求时的处理代码如下：

```
1 private function receiveText($object)
2 {
3     $keyword = trim($object->Content);
4     $category = substr($keyword,0,6);
5     $entity = trim(substr($keyword,6,strlen($keyword)));
6     switch ($category)
7     {
8         case "附近":
9             include("location.php");
10            $location = getLocation($object->FromUserName);
11            if (is_array($location)){
12                include("mapbaidu.php");
13                $content = catchEntitiesFromLocation($entity, $location ["locationX"], $location["locationY"], "5000");
14            }else{
15                $content = $location;
16            }
17            break;
18        default:
19            $content = $object->FromUserName;
20            break;
21    }
22    if(is_array($content)){
23        $result = $this->transmitNews($object, $content);
24    }else{
25        $result = $this->transmitText($object, $content);
26    }
27    return $result;
28 }
```

上述代码简要解释说明如下：

第3行~第5行：分离出用户消息内容中前面2个汉字以后后面的文字。

第6行~第21行：判断前两个汉字是否为“附近”，如果是的话，判断缓存是否有该用户坐标，如果有，则调用百度地图查询数据。

第22行~第27行：根据查询结果是否为数组，若为数组，封装图文消息，若为非数组则封装文本消息。封装后返回结果。

在微信上实现附近搜索后的效果如图7-32所示。

图7-32　附近搜索

7.9　多个功能整合

前面各节的功能都是独立实现的，但大多数时候，我们希望多个功能能同时使用，这就需要对上述功能进行整合。

整合的思路就是通过用户发送的文字提取关键字来判断进入哪一个功能，再调用回复相应的内容。当一个功能不匹配的时候，则进入下一个功能判断。

实现代码如下所示：

```
//接收文本消息
private function receiveText($object)
{
    $keyword = trim($object->Content);
    //判断天气
    if (strstr($keyword, "天气")){
        $city = str_replace('天气', '', $keyword);
        include("weather.php");
        $content = getWeatherInfo($city);
    //判断笑话
    }else if (strstr($keyword, "笑话")){
        include("joke.php");
        $content = getJokeInfo();
    //判断世界杯
    }else if (strstr($keyword, "世界杯")){
        $content[] = array("Title" =>"2014年巴西世界杯赛程","Description" =>"", "PicUrl"
=>"http://images.cnitblog.com/i/340216/201406/111304544204656.jpg", "Url" =>"http://url.cn/RInu1v");
    //其他默认回复
    }else{
        $content = date("Y-m-d H:i:s",time())."\n技术支持 方倍工作室";
    }
    if(is_array($content)){
        if (isset($content[0]['PicUrl'])){
            $result = $this->transmitNews($object, $content);
        }else if (isset($content['MusicUrl'])){
            $result = $this->transmitMusic($object, $content);
        }
    }else{
        $result = $this->transmitText($object, $content);
    }
    return $result;
}
```

上述代码使用if-else-if这样的分支语句实现类别区分，比如发送“深圳天气”之后，代码

```
if (strstr($keyword, "天气")){
```

判断文字中包括“天气”二个字，就进入了天气类别，剩下还要把“深圳”两个字提取出来，这里使用字符串替换的方式，把“天气”两个字替换成空（相当于删除）。代码如下所示：

```
$city = str_replace('天气', '', $keyword);
```

这样就得到了城市名称。

再将天气查询文件包含进来，并且将城市名称传入，如：

```
include("weather.php");
$content = getWeatherInfo($city);
```

这样就查询到了深圳的天气预报信息。

如果没有“天气”两个字，则进行其他的功能判断，处理的方式和天气查询基本一致。

通过上述方法，可以将多个功能进行整合，整合后可以同时实现多个功能的查询。

7.10　本章小结

本章首先分别介绍了天气预报、交通信息查询、中英翻译、空气质量、股票行情及分析、苹果产品信息查询、历史上的今天、附近搜索等功能的实现，然后介绍了如何将各个功能进行整合的方法。

第8章　娱乐类应用开发

休闲娱乐一直是人们生活中最受欢迎的项目，在微信上也不例外，本章将介绍微信中娱乐类应用的开发及实现方法，这些功能包括：笑话、星座运势、周公解梦、姓名测试、夫妻相、在线点歌、一战到底、智能聊天机器人。

8.1　笑话

笑话是人生之必需，是健康生活的调味品、维生素，广泛存在于社会，无处不在。凡是有人群的地方，都离不开它。下面我们介绍如何在微信公众平台中开发笑话功能。

8.1.1　数据获取

目前笑话主要存在于一些门户或者专业的网站，可以使用爬虫程序将其采集下来，然后保存在本地数据库中。

笑话数据库脚本内容如下面SQL脚本所示：

```
DROPTABLEIFEXISTS 'joke';
CREATETABLEIFNOTEXISTS 'joke' (
  'id' int(5) NOTNULL AUTO_INCREMENT,
  'content' varchar(600  ) NOTNULL,
PRIMARYKEY ('id')
) ENGINE=MyISAM  DEFAULT CHARSET=gbk AUTO_INCREMENT=1001 ;
INSERTINTO 'joke' ('id', 'content') VALUES
(1, '她:"因为别人都不同情你,我才做你的妻子。"他:"你总算成功了。现在每个人都因此同情我。"'),
(2, '女:"为什么从前你对我百依百顺,可结婚才三天,你就跟我吵了两天的架?"男:"因为我的忍耐是有限度的。"'),
(3, '燕尔新婚,新娘对新郎说:"今后咱们不兴说'我的'了,要说'我们的'。"新郎去洗澡,良久不出,新娘问:"你在干什么哪?""亲爱的,我在刮我们的胡子呢。"'),
```

该表包含两个字段，id和content，其中id表示序号，content为笑话内容。将该数据文件导入到数据库中，就在本地服务器上实现了一个小型笑话库。

8.1.2　开发实现

在数据库中查询笑话内容的代码如下：

```
function getJokeInfo()
{
    if(isset($_SERVER['HTTP_APPNAME'])){        //SAE
        $mysql_host = SAE_MYSQL_HOST_M;
        $mysql_host_s = SAE_MYSQL_HOST_S;
        $mysql_port = SAE_MYSQL_PORT;
        $mysql_user = SAE_MYSQL_USER;
        $mysql_password = SAE_MYSQL_PASS;
        $mysql_database = SAE_MYSQL_DB;
    }else{
        $mysql_host = "127.0.0.1";
        $mysql_host_s = "127.0.0.1";
        $mysql_port = "3306";
        $mysql_user = "root";
        $mysql_password = "root";
        $mysql_database = "weixin";
    }

    $mysql_table = "joke";
    $id = rand(1, 1000);
    $mysql_state = "SELECT * FROM `".$mysql_table."` WHERE `id` = '".$id."'";
    $con = mysql_connect($mysql_host.':'.$mysql_port, $mysql_user, $mysql_password);
    if (!$con){
```

```
24         die('Could not connect: ' . mysql_error());
25     }
26 
27     mysql_query("SET NAMES 'UTF8'");
28     mysql_select_db($mysql_database, $con);
29     $result = mysql_query($mysql_state);
30 
31     $joke = "";
32     while($row = mysql_fetch_array($result))
33     {
34         $joke = $row["content"];
35         break;
36     }
37     mysql_close($con);
38     return $joke;
39 }
```

上述代码解读如下：

第3行~第17行：定义数据库环境变量，此处判断程序运行环境是SAE还是本地服务器环境，分别定义不同的环境变量。

第19行~第21行：构造了执行查询的SQL语句。从表“joke”中随机查询一条记录。

第23行~第25行：进行数据库连接。如果连接失败，将执行“die”部分。

第27行：设置字符集为“UTF8”，以便正常显示中文。

第28行：设置设置活动的MySQL数据库。

第29行：使用mysql_query（）函数向MySQL发送命令。

第31行~第36行：将查询到的结果存在变量中。

第37行：关闭数据库连接。

第38行：返回查询到的内容。

一个微信查看笑话的功能如图8-1所示。

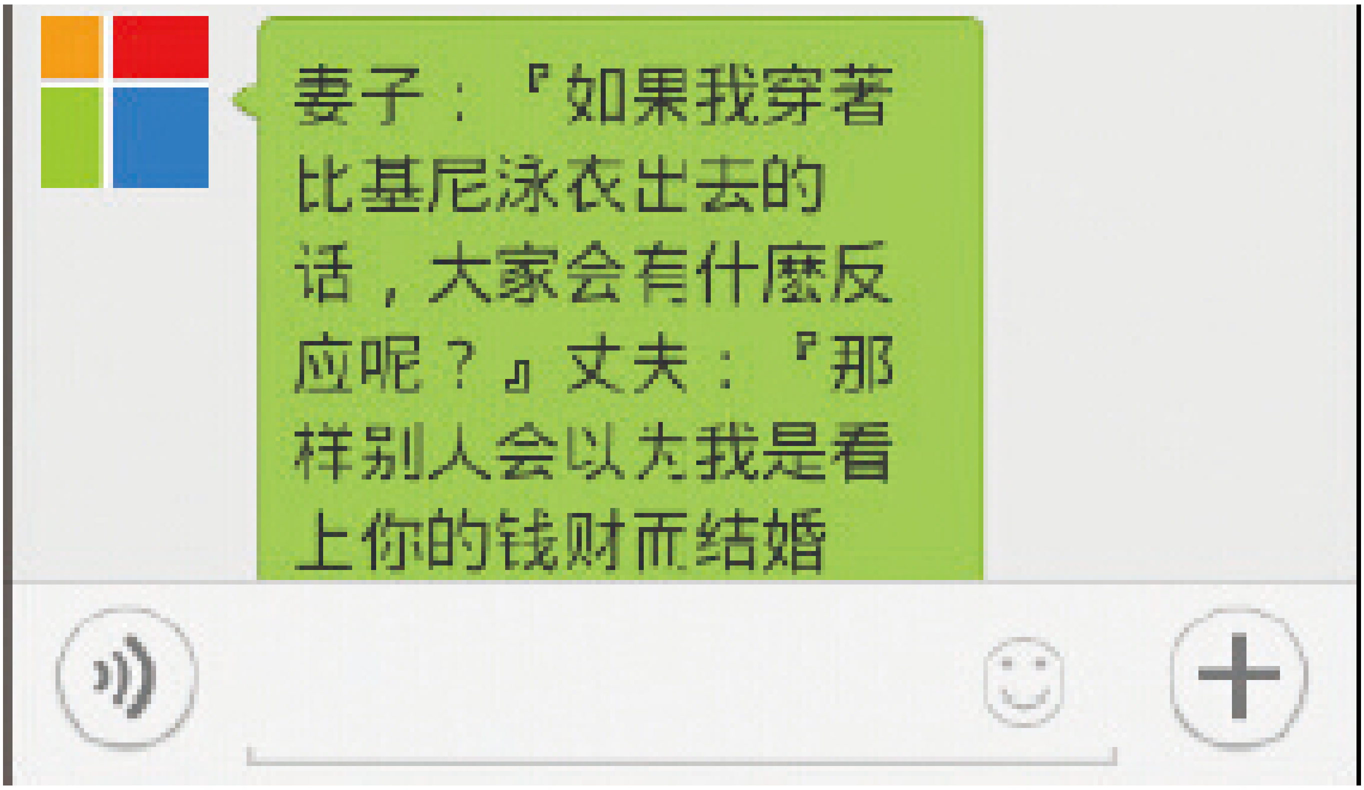


图8-1　笑话

8.2　星座运势

星座运势根据一个人出生日期（阳历）所属的星座，来判定该星座在每一个时间段（一般为日、周、月、年等）发生事情的吉凶。本节介绍在微信公众平台上实现星座运势查询的功能。

8.2.1　新浪星座

手机新浪网提供星座运势查询服务，地址如下所示：

```
http://dp.sina.cn/dpool/astro/starent/starent.php
```

其实际查询页面的一个地址如下：

```
http://dp.sina.cn/dpool/astro/starent/starent.php?type=day&ast=11&vt=4
```

该页面中各参数说明如表8-1所示。

表8-1　星座参数

参数	说明
type	周期类型：今日为 day，明日为 tomorrow，本周为 week，本月为 month
ast	星座数字 id：1：白羊座；2：金牛座；3：双子座；4：巨蟹座；5：狮子座；6：处女座；7：天秤座；8：天蝎座；9：射手座；10：摩羯座；11：水瓶座；12：双鱼座
version	版本，目前为 4

8.2.2　开发实现

根据上述接口，开发一个根据星座名查询运势的功能，相应的代码如下：

```
function getAstrologyInfo($entity)
{
    $capitals = array(
        '白羊座' => '1',
        '金牛座' => '2',
        '双子座' => '3',
        '巨蟹座' => '4',
        '狮子座' => '5',
        '处女座' => '6',
        '天秤座' => '7',
        '天蝎座' => '8',
```

```
12          '射手座' => '9',
13          '摩羯座' => '10',
14          '水瓶座' => '11',
15          '双鱼座' => '12'
16      );
17      if (!array_key_exists($entity, $capitals))
18      {
19          return "星座名只有以下这些：\n白羊座 金牛座 双子座 巨蟹座 狮子座 处女座 天秤座 天蝎座 射手座 摩羯座 水瓶座 双鱼座";
20      }
21      $astrologyArray[] = array(
22      "Title" =>$entity."运势",
23      "Description" =>"",
24      "PicUrl" =>"",
25      "Url" =>"http://dp.sina.cn/dpool/astro/starent/starent.php?type=day&ast=".$capitals[$entity]."&vt=4");
26
27      return $astrologyArray;
28  }
```

上述代码解读如下：

第3行~第16行：定义星座名称和相应的id号。

第17行~第19行：判断输入是否为星座名称。

第21行~第25行：构造一个单图文消息，标题为星座名，链接为对应星座运势的新浪手机版网页。

星座运势的运行效果如图8-2所示。

图8-2 星座

8.3 周公解梦

梦是人类精神生活的一种很特殊的现象。人对梦的思考，属于人对自身的一种审视，因而也是一种自我认识。梦，是窥探内心的一面隐秘之镜，是另一种虚幻却真实的人生体验。本节介绍在微信公众平台上实现周公解梦的功能。

8.3.1 数据获取

提供周公解梦的网站很多，可以发现能解的数据在5000个左右，我们可选择一家网站将其内容收集下来做成自己的数据库。

周公解梦数据表结构及内容如下面SQL脚本所示：

```
DROPTABLEIFEXISTS 'dream';
CREATETABLEIFNOTEXISTS 'dream' (
  'id' int(4) NOTNULL AUTO_INCREMENT,
  'title' varchar(20) NOTNULL,
  'content' varchar(1024) NOTNULL,
PRIMARYKEY ('id')
) ENGINE=InnoDB  DEFAULT CHARSET=gbk AUTO_INCREMENT=4488 ;
INSERTINTO 'dream' ('id', 'title', 'content') VALUES
(1, '父母', '梦见自己成了幼儿与父母相处幸运的事情即将发生。可以尝到美味可口的咖啡、得到电影招待券等等。你将不觉莞尔一笑。梦见和父母快乐地有说有笑家中将起波澜。对父母感觉厌烦,会有离家出走的行为。你还未
(2, '护士', ' 已婚女子梦见护士不久会怀孕,得贵子。少女梦见有一群美貌的护士很快要出嫁。少女梦见和护士争吵婚事不顺利,迟迟不能出嫁。病人梦见护士痛苦很快要过去。'),
(3, '老师', ' 梦见受老师称赞在学业方面乌云密布。由于连日的熬夜,在课堂上竟开始打盹结果受到老师批评。这就是所谓反梦。梦见受老师责骂家人关系极佳。对双亲能克尽孝道,你将令人刮目相看。也许每个月的零用钱会大
(4, '男孩', '梦见男孩,是祥兆。女人梦见男孩,会生病。梦见漂亮的男孩,朋友会忘恩负义。梦见生男孩,生活会幸福、恬适。梦见男孩夭亡,大难将降临。女人梦见自己将要生一个男孩,意味着会过上幸福舒畅的生活'),
```

将该数据文件导入到数据库中，就可以直接在本地查询获得结果。

8.3.2 开发实现

当公众账号接收到用户发送“梦见***”的文本消息的时候，相应的处理代码如下：

```
private function receiveText($object)
{
    $keyword = trim($object->Content);
    $result = "";
    $content = "";

    if (substr($keyword, 0, 6) == "梦见"){
        $keyword = trim(substr($keyword, 6, strlen($keyword)));
        include("dream.php");
        $content = getDreamInfo($keyword);
    }

    $result = $this->transmitText($object, trim($content));
    return $result;
}
```

上述代码中，先判断前两个字是否为“梦见”，如果是的话，取出后面的文字，并且传替到获取解梦结果的函数中，获得内容后，再封装文本消息返回。

而解梦的程序实现如下：

```
1  functiongetDreamInfo($content)
2  {
3  if ($content == ""){
4  return "梦见什么？";
5      }
6
7  if(isset($_SERVER['HTTP_APPNAME'])){        // SAE
8       $mysql_host = SAE_MYSQL_HOST_M;
9       $mysql_host_s = SAE_MYSQL_HOST_S;
10      $mysql_port = SAE_MYSQL_PORT;
11      $mysql_user = SAE_MYSQL_USER;
12      $mysql_password = SAE_MYSQL_PASS;
13      $mysql_database = SAE_MYSQL_DB;
14 }else{
15      $mysql_host = "127.0.0.1";
16      $mysql_host_s = "127.0.0.1";
17      $mysql_port = "3306";
18      $mysql_user = "root";
19      $mysql_password = "root";
20      $mysql_database = "weixin";
21     }
22
23 $mysql_table = "dream";
24 $mysql_state= "SELECT * FROM '".$mysql_table."' WHERE 'title' LIKE '%".
$content."%'";
25 $con = mysql_connect($mysql_host_s.':'.$mysql_port, $mysql_user, $mysql_
password);
26 if (!$con){
27 die('Could not connect: ' . mysql_error());
28     }
29
30 mysql_query("SET NAMES 'UTF8'");
31 mysql_select_db($mysql_database, $con);
32 $result = mysql_query($mysql_state);
33 $contentStr = "周公睡觉去了";
34 while($row = mysql_fetch_array($result))
35     {
36 if (isset($row["content"]) && ($row["content"] != "")){
37      $contentStr = $row["content"];
38         }
39 break;
40     }
41      mysql_close($con);
42      return$contentStr;
43 }
```

上述代码解读如下：

第3行~第5行：判断传入内容是否为空。

第7行~第21行：定义数据库环境变量。

第23行~第24行：构造了执行查询的SQL语句。

第25行~第32行：执行数据库查询。

第33行：定义默认回复内容。

第34行~第40行：从查询结果数组中取出内容。

第41行~第42行：关闭数据库连接，返回结果。

周公解梦的实现效果如图8-3所示。

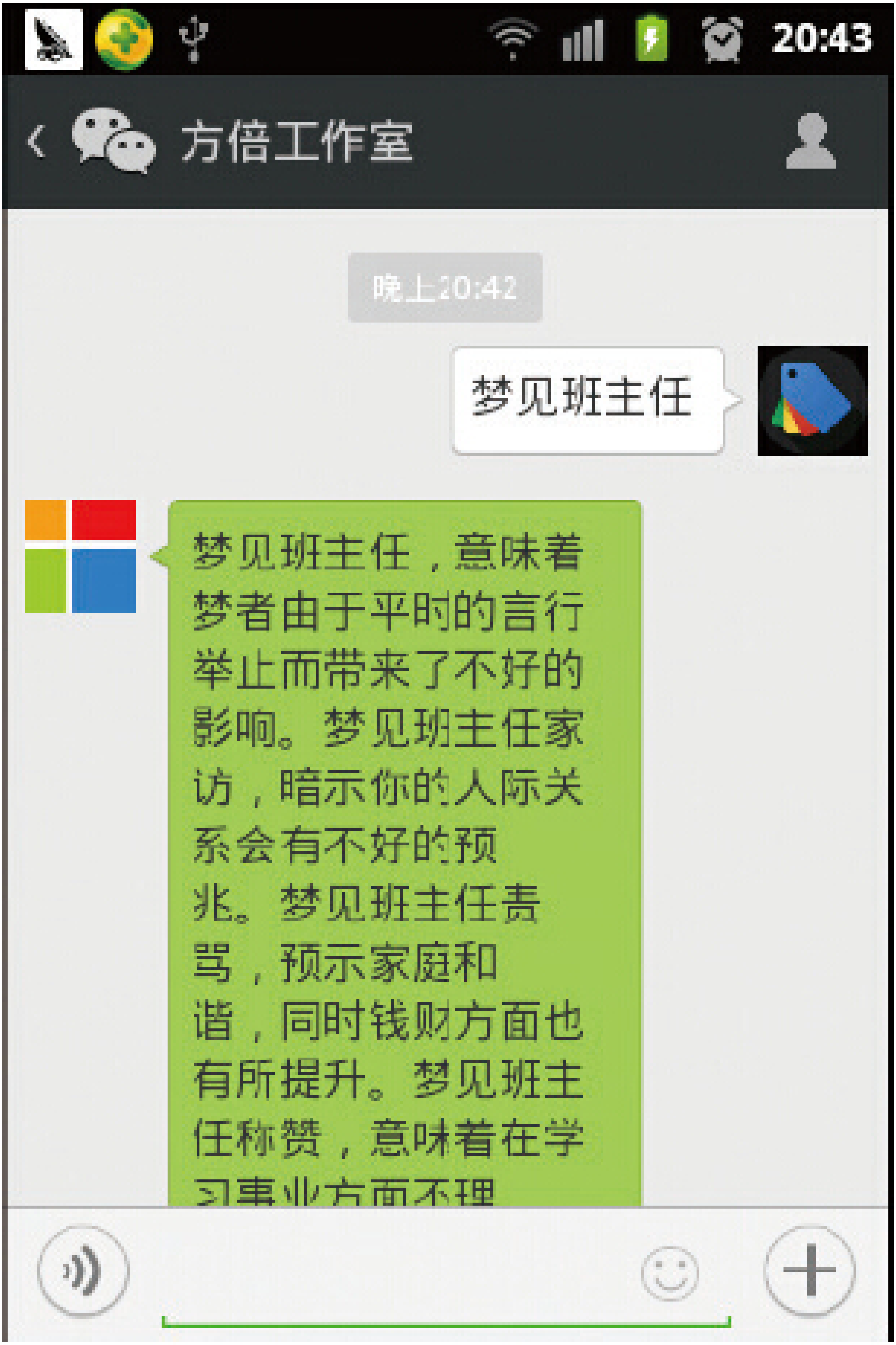

20:43
方倍工作室
晚上20:42
梦见班主任
梦见班主任，意味着梦者由于平时的言行举止而带来了不好的影响。梦见班主任家访，暗示你的人际关系会有不好的预兆。梦见班主任责骂，预示家庭和谐，同时钱财方面也有所提升。梦见班主任称赞，意味着在学

图8-3　周公解梦

8.4　姓名测试

姓名测试是指根据姓名的笔画数以及一定规则数理关系，运用阴阳五行相生相克的道理，来预测姓名的吉凶好坏。本节介绍如何在微信中实现姓名测试的功能。

8.4.1　数据获取

姓名测试打分手机版网站提供姓名测试服务，地址为

```
http://m.1518.com/
```

经过试验与分析，其实际查询接口地址如下：

```
http://m.1518.com/xingming_view.php?word=%D5%C5%C8%FD&submit1=%C8%B7%B6%A8&FrontType=1
```

该接口各参数说明如表8-2所示。

表8-2　姓名测试接口参数

参数	说明
word	姓名。使用 GB2312 的编码，传递时需要做 urlencode
submit	“确定”在 GB2312 编码下的 urlencode
FrontType	1

上述接口在浏览器中打开如图8-4所示。

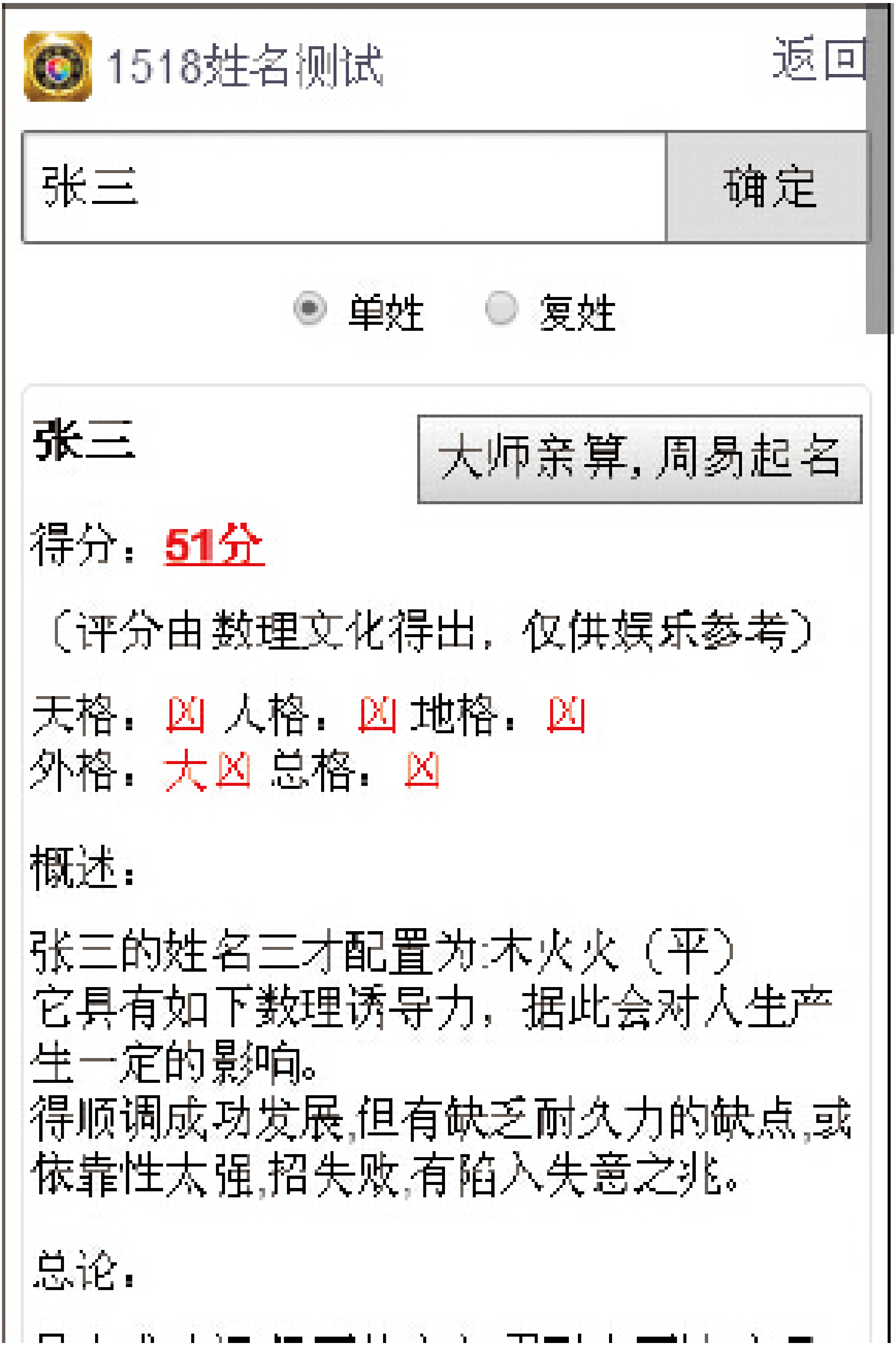

1518姓名测试
返回
张三
确定
单姓
复姓
张三
大师亲算，周易起名
得分：51分
（评分由数理文化得出，仅供娱乐参考）
天格：凶 人格：凶 地格：凶
外格：大凶 总格：凶
概述：
张三的姓名三才配置为:木火火（平）
它具有如下数理诱导力，据此会对人生产生一定的影响。
得顺调成功发展,但有缺乏耐久力的缺点,或依靠性太强,招失败,有陷入失意之兆。
总论：

图8-4　姓名测试

查看该网页源代码，内容部分代码显示如下：

```
<div id="detail">
  <h3>张三<span><a href=
"qiming/"><img src="images/bbqm_shouji_small.jpg" border="0"></a></span></h3>
  <dl>
    <dt>得分：<u><strong>51分</strong></u></dt>
    <dd>
      <div>（评分由数理文化得出，仅供娱乐参考）</div>
    </dd>
    <dd>天格：<font color=red>凶</font>
      人格：<font color=red>凶</font>
      地格：<font color=red>凶</font><br>
      外格：<font color=red>大凶</font>
      总格：<font color=red>凶</font>
    </dd>
  </dl>
  <dl>
    <dt>概述：</dt>
    <dd>
      张三的姓名三才配置为:木火火（平）<br>它具有如下数理诱导力，据此会对人生产生一定的影响。<br>
    得顺调成功发展,但有缺乏耐久力的缺点,或依靠性太强,招失败,有陷入失意之兆。      </dd>
  </dl>
  <dl>
    <dt>总论：</dt>
    <dd>虽有成功运,但不甘寂寞,忍耐力不够,容易一意孤行,陷入孤军奋斗,如能修身养性,定能成功顺调。名利双利,若天运五行属水,青年时代事业难成又逢灾,老运亦欠佳。      </dd>
  </dl>
  <dl>
    <dt>性格：</dt>
    <dd>
    外表乐观而豪爽,交际应酬有一套,人缘又好,但个性喜怒无常,感情易热易冷,个性急燥,容易意气用事。</dd>
  </dl>
```

可以看到，事件内容都放置在div区块下的dl标签中。我们可以通过循环遍历的方式将相应数据取出来。

8.4.2　开发实现

我们使用simple_html_dom来抓取所需要的页面，相应代码如下：

```
 1 function getFortuneInfo($keyword)
 2 {
 3     if ($keyword == ""){
 4         return "请发送姓名+人名，例如"姓名张三"";
 5     }
 6     if ((strlen($keyword) < 6) || (strlen($keyword) > 12)){
 7         return "人名长度为2到4个汉字";
 8     }
 9     try {
10         include('simple_html_dom.php');
11         $url = "http://m.1518.com/xingming_view.php?word=".urlencode (mb_convert_encoding($keyword, 'gb2312', 'utf-8')). "&submit1=%C8%B7%B6%A8&FrontType=1";
12         $html_fotune = file_get_html($url);
13         if (!isset($html_fotune)){
14             $html_fotune->clear();
15             return "程序检索出错！\n如果经常这样，请联系方倍工作室。";
16         }
17         $infomation = "";
18         foreach($html_fotune->find('div[id="detail"] dl') as $item) {
19             $curText = $item->plaintext;
20             $curText = preg_replace('/\s{2,}/i', ' ', $curText);
21             $curText = preg_replace('/\t{2,}/i', ' ', $curText);
22             $curText = preg_replace('/: \s/i', ': ', $curText);
23             $infomation .= trim($curText)."\n";
24             if (strlen($infomation) > 2000){break;}
25         }
26         $html_fotune->clear();
27         $infomation = str_replace("\r\n", "\n", $infomation);
28         $infomation = str_replace("天格", "\n天格", $infomation);
29         $infomation = str_replace("  ", "", $infomation);
30         return trim($infomation);
31     }catch (Exception $e){
32 
33     }
34 }
```

上述代码解读如下：

第3行~第8行：判断姓名是否为空以及长度值。

第10行：引用HTML解析器库文件。

第11行：定义要访问的网址URL。使用函数mb_convert_encoding（）将“张三”从UTF-8编码转换为GB2312编码，再将字符串以URL编码。

第12行：从页面地址中创建文档对象模型。

第13行~第16行：判断模型是否创建成功。

第17行：定义变量，用于保存最终返回内容。

第18行~第25行：查找id为“detail”的div元素，并在该元素下找dl元素，符合这样条件的均为解析内容，通过循环遍历的方式来获取，取其纯文本内容。使用正则表达式替换掉空格等内容。最后，判断长度是否超过2000，若超过则直接退出。

第26行：清理文档对象，释放资源。

第27行~第29行：使用字符串替换等方式，继续删除空格等内容。

第30行：返回文本消息。

姓名测试的实现效果如图8-5所示。

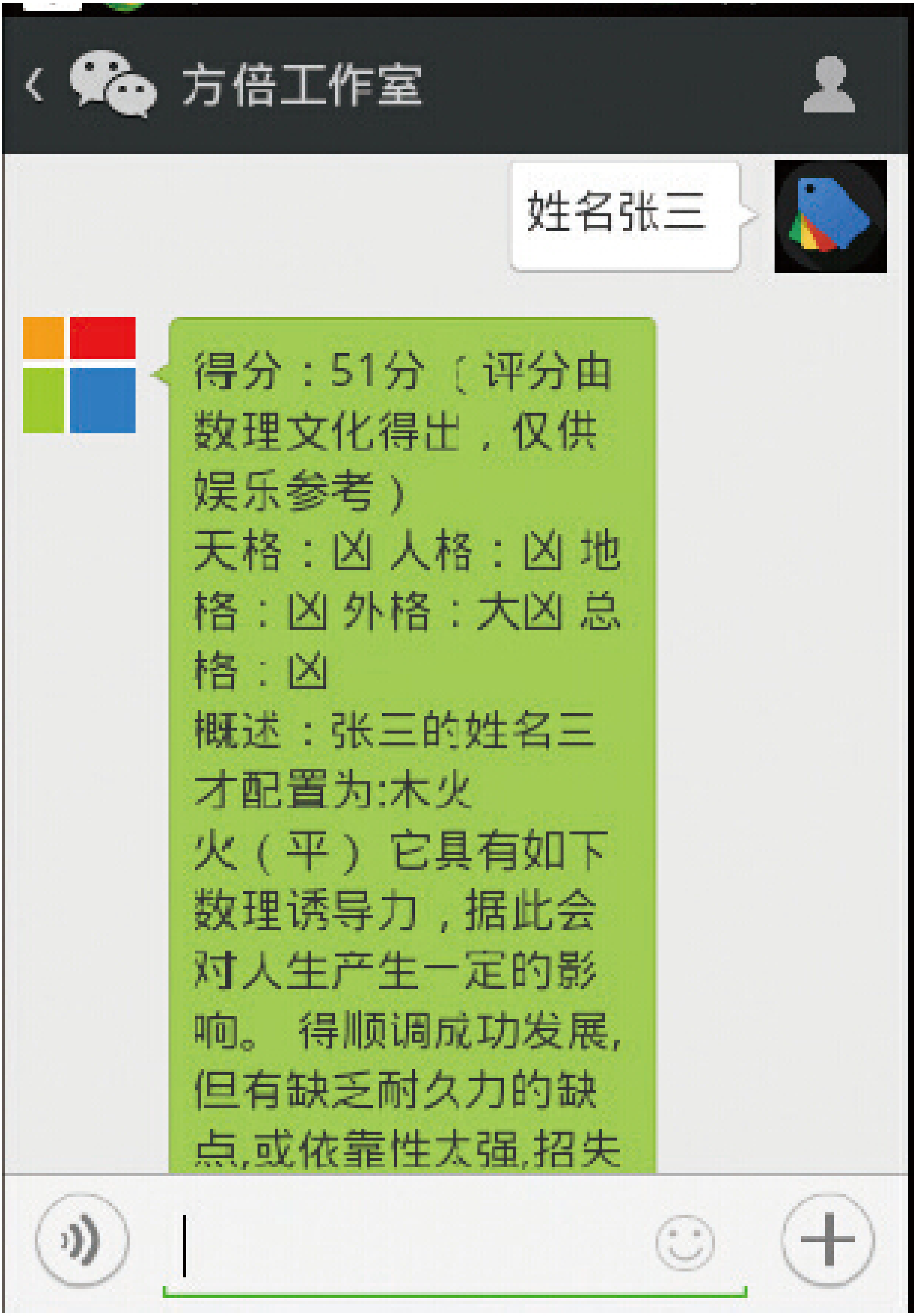

方倍工作室
姓名张三
得分：51分（评分由数理文化得出，仅供娱乐参考）
天格：凶 人格：凶 地格：凶 外格：大凶 总格：凶
概述：张三的姓名三才配置为:木火火（平）它具有如下数理诱导力，据此会对人生产生一定的影响。 得顺调成功发展,但有缺乏耐久力的缺点,或依靠性太强,招失

图8-5　姓名测试

8.5 夫妻相

在一起生活多年的夫妻中，有很多人在外貌、表情、形体，甚至许多爱好上都有非常相像的地方。有的单从面相上来看竟惊人的相似，如同一家人，有人将此称为“夫妻相”。著名的婚恋交友网站“世纪佳缘”也推出过夫妻相这一功能。本节介绍如何在微信公众平台上实现夫妻相功能。

8.5.1 人脸识别

Face++是北京旷视科技（Megvii）有限公司旗下的新型视觉服务平台，旨在提供简单易用、功能强大、平台兼容的新一代视觉服务。

Face++团队专注于研发世界最好的人脸检测、识别、分析和重建技术，通过融合机器视觉、机器学习、大数据挖掘及3D图形学技术，致力于将最新、性能最好、使用最方便的人脸技术提供给广大开发者和用户。通过提供云端API、离线SDK以及面向用户的自主研发产品形式，将人脸识别技术广泛应用到互联网及移动应用场景中。

我们使用Face++的接口来完成这一功能。使用接口时需要先注册一个Face++账号，账号登录后需要先创建一个应用，如图8-6所示。

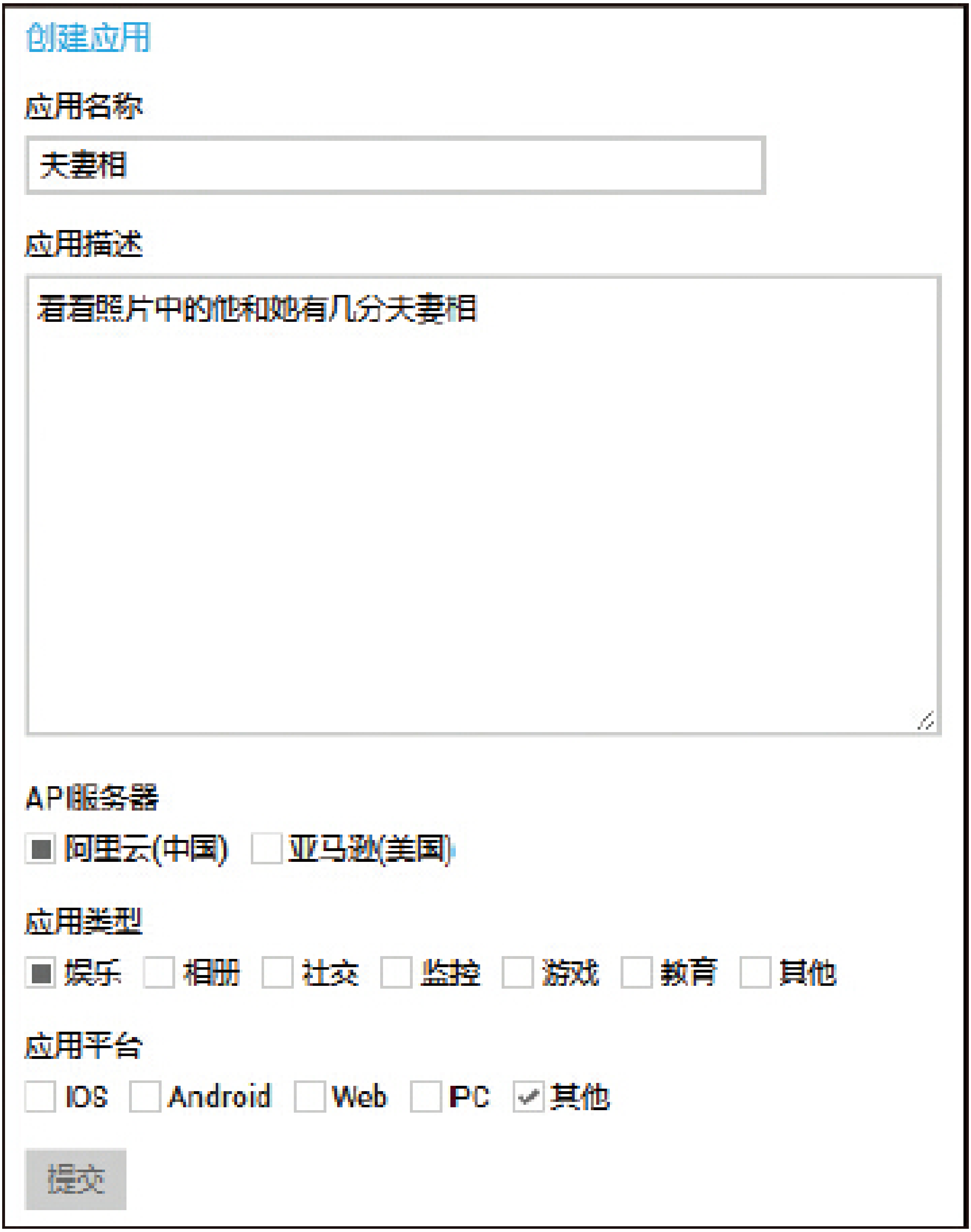


图8-6　创建Face++应用

选择API服务器的时候，需要注意国内用户一般选择“阿里云（中国）”，其他内容可以自由填写或选择。提交后，我们将得到API Key和API Secret，如图8-7所示。

图8-7　创建夫妻相应用

Face++接口的功能一共分为6大类：人脸检测与分析、训练模型、人脸识别、人脸聚类与分组、person管理、faceSet管理、group管理、信息查询。详细接口功能如表8-3所示。开发者可以使用这些接口开发出更多有趣的人脸相关的应用。

表8-3　Face++API列表

接口分类	接口名称	功能说明
detect	/detection/detect	检测一张照片中的人脸信息（脸部位置、年龄、种族、性别等）
	/detection/landmark	检测给定人脸相应的面部轮廓、五官等关键点的位置
train	/train/verify	调用 /recognition/verify 之前需要运行的训练
	/train/search	调用 /recognition/search 之前需要运行的训练
	/train/identify	调用 /recognition/identify 之前需要运行的训练
recognition	/recognition/compare	对比两张人脸的相似程度
	/recognition/verify	给定人脸和人，判断这张人脸是否是给定的人
	/recognition/search	给定人脸和人脸集合，在集合中找最相似的人脸
	/recognition/identify	给定人脸和人群，找到人群中最像这张脸的人
grouping	/grouping/grouping	给定人脸集合，将集合中的人脸分成几类，每一类为同一个人
person	/person/create	创建一个人
	/person/delete	删除一个人
	/person/add_face	向一个人添加一张人脸
	/person/remove_face	从一个人中删除一张人脸
	/person/set_info	为一个人设定备注等信息
	/person/get_info	获得一个人的备注等信息
faceset	/faceset/create	创建一个人脸集合
	/faceset/delete	删除一个人脸集合
	/faceset/add_face	向一个人脸集合添加人脸
	/faceset/remove_face	从一个人脸集合中删除人脸
	/faceset/set_info	为一个人集合设定备注等信息
	/faceset/get_info	获得一个人集合的备注等信息
group	/group/create	创建一个人群
	/group/delete	删除一个人群
	/group/add_person	向人群中添加一个人
	/group/remove_person	从人群中删除一个人
	/group/get_info	获得人群的备注等信息
info	/info/get_image	获取一张图片的信息，包括其中的人脸信息
	/info/get_face	获取一组人脸的信息
	/info/get_person_list	获取该应用中所有的人
	/info/get_faceset_list	获取该应用中所有的人脸集合
	/info/get_group_list	获取该应用中所有的人群
	/info/get_session	获取 session 的状态与结果
	/info/get_app	获取应用的相关信息

8.5.2 实现方案

我们将使用上述接口列表中的两个接口来实现夫妻相功能。第一步，检测照片中的人脸信息，获得两个人的人脸；第二步，将两个人的人脸进行对比，获得他们的相似度。

1.检测人脸信息

检测给定图片（image）中的所有人脸（face）的位置和相应的面部属性接口地址如下：

```
http://apicn.faceplusplus.com/v2/detection/detect
```

该接口相关参数如表8-4所示。

表8-4 检测人脸信息接口参数说明

参数	是否必须	说明
api_key	必须	APP 的 Face++API Key
api_secret	必须	APP 的 Face++API Secret
url 或 img[POST]	必须	待检测图片的 URL 或者通过 POST 方法上传的二进制数据，原始图片大小需要小于 3M
mode	可选	检测模式可以是 normal（默认）或者 oneface。在 oneface 模式中，检测器仅找出图片中最大的一张脸
attribute	可选	可以是 none 或者由逗号分隔的属性列表。默认为 gender、age、race、smiling。目前支持的属性包括：gender、age、race、smiling、glass、pose
tag	可选	可以为图片中检测出的每一张 face 指定一个不包含 ^@，&=*'" 等非法字符且不超过 255 字节的字符串作为 tag，tag 信息可以通过 /info/get_face 查询
async	可选	如果置为 true，该 API 将会以异步方式被调用；也就是立即返回一个 session id，稍后可通过 /info/get_session 查询结果。默认值为 false

该接口调用举例如下：

```
http://apicn.faceplusplus.com/v2/detection/detect?api_key=6efbacb73e7d1bbc424f41fa656c328f&api_secret=eD5ie8tydoBVMwgNlbe7XKUYvTCVs27e&url=http://img3.yxlady.com/yl/UploadFiles
```

上述接口返回结果如下所示：


```
{
    "face": [
        {
            "attribute": {
                "age": {
                    "range": 5,
                    "value": 30
                },
                "gender": {
                    "confidence": 99.5937,
                    "value": "Male"
                },
                "glass": {
                    "confidence": 99.9969,
                    "value": "None"
                },
                "pose": {
                    "pitch_angle": {
                        "value": 0.000011100340000000002
                    },
                    "roll_angle": {
                        "value": -2.79166
                    },
                    "yaw_angle": {
                        "value": 0
                    }
                },
                "race": {
                    "confidence": 98.5664,
                    "value": "Asian"
                },
                "smiling": {
                    "value": 97.5054
                }
            },
            "face_id": "7f24d567a5e557b5853be72c3e5c2134",
            "position": {
                "center": {
                    "x": 60.681818,
                    "y": 48.293515
                },
                "eye_left": {
                    "x": 55.736591,
                    "y": 41.227645
                },
                "eye_right": {
                    "x": 65.533636,
                    "y": 40.510239
                },
                "height": 27.645051,
                "mouth_left": {
                    "x": 56.925909,
                    "y": 54.977133
                },
                "mouth_right": {
                    "x": 65.680909,
                    "y": 54.565529
                },
                "nose": {
                    "x": 60.736364,
                    "y": 50.108532
                },
                "width": 18.181818
            },
            "tag": ""
        },
        {
            "attribute": {
                "age": {
                    "range": 5,
                    "value": 17
                },
                "gender": {
                    "confidence": 99.9781,
                    "value": "Female"
                },
                "glass": {
                    "confidence": 99.9815,
                    "value": "None"
                },
                "pose": {
                    "pitch_angle": {
                        "value": 0.000019753399999999996
                    },
                    "roll_angle": {
                        "value": 1.75177
```

```
                    },
                    "yaw_angle": {
                        "value": 4
                    }
                },
                "race": {
                    "confidence": 99.471,
                    "value": "Asian"
                },
                "smiling": {
                    "value": 87.1365
                }
            },
            "face_id": "c772b4b66c00d46b15344eff74b56e48",
            "position": {
                "center": {
                    "x": 35.568182,
                    "y": 62.286689
                },
                "eye_left": {
                    "x": 31.1675,
                    "y": 55.166553
                },
                "eye_right": {
                    "x": 40.813182,
                    "y": 55.609556
                },
                "height": 26.279863,
                "mouth_left": {
                    "x": 31.192045,
                    "y": 68.601706
                },
                "mouth_right": {
                    "x": 39.490455,
                    "y": 69.341638
                },
                "nose": {
                    "x": 35.8725,
                    "y": 64.405802
                },
                "width": 17.5
            },
            "tag": ""
        }
    ],
    "img_height": 293,
    "img_id": "3005132383841edd08c9b500fb1fe2c4",
    "img_width": 440,
    "session_id": "4e64c73fec19442cbefde3cf9bd6b53d",
    "url": "http://img3.yxlady.com/yl/UploadFiles_5361/20110513/20110513130615793.jpg"
}
```

上述返回结果参数说明如表8-5所示。

表8-5　检测人脸信息接口结果参数说明

参数	类型	说明
session_id	string	相应请求的 session 标识符，可用于结果查询
url	string	请求中图片的 url
img_id	string	Face++ 系统中的图片标识符，用于标识用户请求中的图片
face_id	string	被检测出的每一张人脸在 Face++ 系统中的标识符
img_width	integer	请求图片的宽度
img_height	integer	请求图片的高度
face	array	被检测出的人脸的列表
width	float	0 ~ 100 之间的实数，表示检出的脸的宽度在图片中百分比
height	float	0 ~ 100 之间的实数，表示检出的脸的高度在图片中百分比
center	object	检出的人脸框的中心点坐标，x & y 坐标分别表示在图片中的宽度和高度的百分比（0 ~ 100 之间的实数）
eye_left	object	相应人脸的左眼坐标，x & y 坐标分别表示在图片中的宽度和高度的百分比（0 ~ 100 之间的实数）
eye_right	object	相应人脸的右眼坐标，x & y 坐标分别表示在图片中的宽度和高度的百分比（0 ~ 100 之间的实数）
mouth_left	object	相应人脸的左侧嘴角坐标，x & y 坐标分别表示在图片中的宽度和高度的百分比（0 ~ 100 之间的实数）
mouth_right	object	相应人脸的右侧嘴角坐标，x & y 坐标分别表示在图片中的宽度和高度的百分比（0 ~ 100 之间的实数）
nose	object	相应人脸的鼻尖坐标，x & y 坐标分别表示在图片中的宽度和高度的百分比（0 ~ 100 之间的实数）
attribute	object	包含一系列人脸的属性分析结果
gender	object	包含性别分析结果，value 的值为 Male/Female，confidence 表示置信度
age	object	包含年龄分析结果，value 的值为一个非负整数表示估计的年龄，range 表示估计年龄的正负区间
race	object	包含人种分析结果，value 的值为 Asian/White/Black，confidence 表示置信度
smiling	object	包含微笑程度分析结果，value 的值为 0 ~ 100 的实数，越大表示微笑程度越高
glass	object	包含眼镜佩戴分析结果，value 的值为 None/Dark/Normal，confidence 表示置信度
pose	object	包含脸部姿势分析结果，包括 pitch_angle、roll_angle、yaw_angle，分别对应抬头、旋转（平面旋转）、摇头。单位为角度

2.比较人脸相似度

计算两个Face的相似性以及五官相似度接口地址如下：

```
https:// apicn.faceplusplus.com/v2/recognition/compare
```

该接口的相关参数如表8-6所示。

表8-6　计算相似度接口参数说明

参数	是否必须	说明
api_key	必须	APP 的 Face++API Key
api_secret	必须	APP 的 Face++API Secret
face_id1	必须	第一个 face 的 face_id
face_id2	必须	第二个 face 的 face_id
async	可选	如果置为 true，该 API 将会以异步方式被调用；也就是立即返回一个 session id，稍后可通过 /info/get_session 查询结果。默认值为 false

该接口调用举例如下：

```
https:// apicn.faceplusplus.com/v2/recognition/compare?api_key=6efbacb73e7d1bbc424f41fa656c328f&api_secret=eD5ie8tydoBVMwgNlbe7XKUYvTCVs27e&face_id2=7f24d567a5e557b5853be72c3e5
```

返回结果列表如下所示：

```
{
    "component_similarity": {
        "eye": 85.802307,
        "eyebrow": 72.329025,
        "mouth": 89.68277,
        "nose": 55.381519
    },
    "session_id": "2cc4e8d04e28466396bde8b83132205b",
    "similarity": 51.831638
}
```


上述返回结果参数说明如表8-7所示。

表8-7　计算相似度接口结果参数说明

参数	类型	说明
component_similarity	object	包含人脸中各个部位的相似性，目前包含 eyebrow(眉毛)、eye(眼睛)、nose(鼻子）与 mouth（嘴）的相似性。每一项的值为一个 0 ～ 100 之间的实数，表示相应部位的相似性
similarity	float	一个 0 ～ 100 之间的实数，表示两个 face 的相似性

有了上述两个接口，就能实现检测照片中男女夫妻相的功能了。

8.5.3　开发实现

在基础接口中我们曾讲到，当用户发送图片给公众账号的时候，微信公众账号将收到一个图片消息，该消息中包含PicUrl和MediaId两项参数，分别表示图片链接和图片消息媒体id。

在这里，将直接把图片链接地址提交给Face++来处理。

Face++的类定义如下：

```
1 class FacePlusPlus
2 {
3     private $api_server_url;
4     private $auth_params;
5
6     public function __construct()
7     {
8         $this->api_server_url = "http://apicn.faceplusplus.com/";
9         $this->auth_params = array();
10        $this->auth_params['api_key'] = "6efbacb73e7d1bbc424f41fa656c328f";
11        $this->auth_params['api_secret'] = "eD5ie8tydoBVMwgNlbe7XKUYvTCVs27e";
12    }
13
14    //人脸检测
15    public function face_detect($urls = null)
16    {
17        return $this->call("detection/detect", array("url"=>$urls));
18    }
19
20    //人脸比较
21    public function recognition_compare($face_id1, $face_id2)
22    {
23        return $this->call("recognition/compare", array("face_id1"=> $face_id1, "face_id2"=>$face_id2));
24    }
25
26    protected function call($method, $params = array())
27    {
28        $url = $this->api_server_url."$method?".http_build_query (array_merge($this->auth_params, $params));
29        $ch = curl_init();
30        curl_setopt($ch, CURLOPT_URL, $url);
31        curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
32        $data = curl_exec($ch);
33        curl_close($ch);
34        $result = json_decode($data);
35        return $result;
36    }
37 }
```

上述代码定义了FacePlusPlus类，在类中定义了两个成员$api_server_url和$auth_params以及三个方法face_detect()、recognition_compare()、call()；前两个方法分别定义了人脸检测接口和人脸比较接口的实现。

第3行~第4行：定义了两个成员变量$api_server_url和$auth_params。

第6行~第12行：类的构造函数，在构造函数中给成员属性定义值，包括接口服务器的url、API Key和API Secret。

第14行~第18行：定义人脸检测接口的方法。

第20行~第24行：定义人脸比较接口的方法。

第26行~第36行：定义接口调用函数，内部使用cURL实现。

获取图片识别结果的代码如下所示：

```
function getImageInfo($url)
{
    $faceObj = new FacePlusPlus();
    $detect = $faceObj->face_detect($url);
    $numbers = isset($detect->face)? count($detect->face):0;
    if (($detect->face[0]->attribute->gender->value != $detect-> face[1]->attribute->gender->value) && $numbers == 2){
        $compare = $faceObj->recognition_compare($detect->face[0]-> face_id,$detect->face[1]->face_id);
      $result = getCoupleComment($compare->component_similarity-> eye, $compare->component_similarity->mouth, $compare->component_similarity->nose, $compare->component_simil
        return $result;
    }else{
        return "似乎不是一男一女，无法测试夫妻相";
    }
}
```

在上述代码中，先使用人脸检测接口获得图片的识别结果，再判断结果中是否得到两个性别不同的人，如果返回结果不是两个人脸且性别不同，则提示无法测试夫妻相，否则将结果中的face_id进行比较，获得相似度的结果。

获得结果以后，我们需要对结果进行加工，以让功能更贴近实际情况，相应的代码如下：

```
function getCoupleComment($eye, $mouth, $nose, $eyebrow, $similarity)
{
    $index = round(($eye + $mouth + $nose + $eyebrow) / 4);
    if ($index < 40){
        $comment = "花好月圆";
    }else if ($index < 50){
        $comment = "相濡以沫";
    }else if ($index < 60){
        $comment = "情真意切";
    }else if ($index < 70){
        $comment = "郎才女貌";
    }else if ($index < 80){
        $comment  = "心心相印";
    }else if ($index < 90){
        $comment  = "浓情蜜意";
    }else{
        $comment = "海誓山盟";
    }
    return "【夫妻相指数】\n得分：".$index."\n评语：".$comment;
}
```

上述代码中，将获取眼睛、嘴巴、眉毛、鼻子4个部分相似值的平均值作为最终相似指数，并且根据指数大小，添加表现夫妻关系的评语。

这样，借助强大的Face++人脸识别接口，一个夫妻相功能就实现了，效果如图8-8所示，图中人脸经过处理。

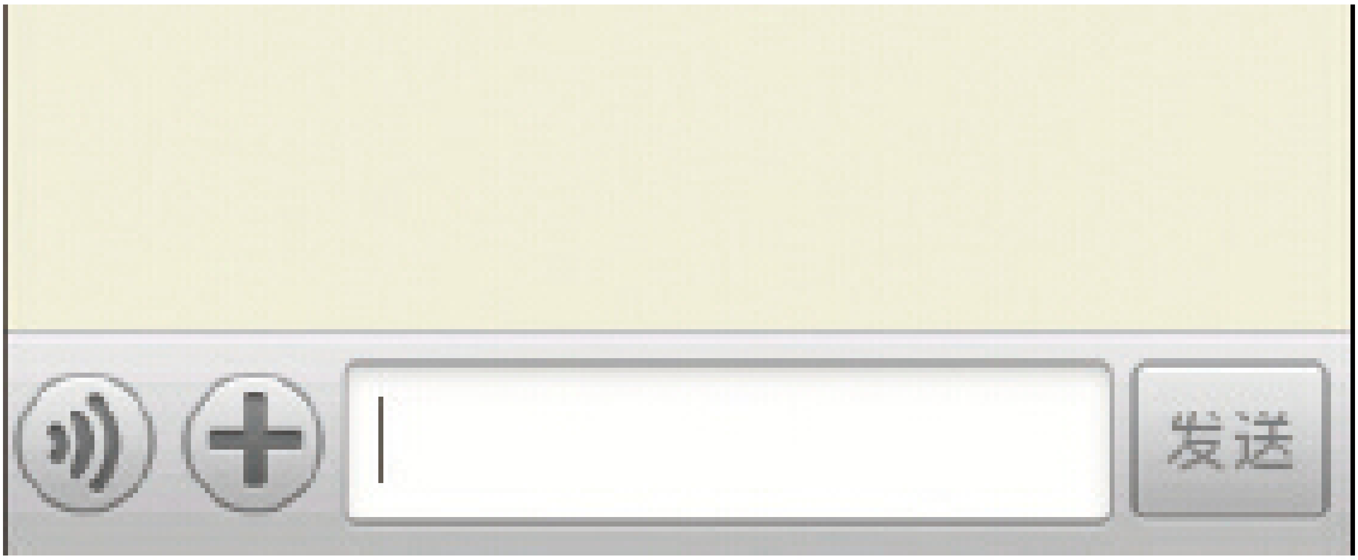


图8-8　夫妻相测试

8.6　在线点歌

微信公众平台在2012年年底开放了音乐消息接口，基于这一接口，我们可以做出微信公众平台上的点歌功能。

8.6.1　音乐掌门人

百度音乐掌门人有一个未公开的MP3音乐接口，可以通过歌手名称及歌曲名称搜索音乐，其接口地址如下：

```
http://box.zhangmen.baidu.com/x?op=12&count=1&title=最炫民族风$$凤凰传奇$$$$
```

也可以省略歌手名称，只填写歌曲名称，示例如下：

```
http://box.zhangmen.baidu.com/x?op=12&count=1&title=最炫民族风$$
```

不指定歌手名称的时候，返回的结果质量稍差。该接口中，“凤凰传奇”为歌手名称，“最炫民族风”为歌曲名称。

上述接口返回的XML内容如下所示：

```
<result>
  <count>5</count>
  <url>
    <encode>
      <![CDATA[
      http://zhangmenshiting.baidu.com/data2/music/84995052/aWZsbWpmbGpfn6NndK6ap5WXcGVmm5tplWdmmGVtbGmSa5mWbZxsaZJpl2aWZ21wapSXZmWYcJplZWhlmJhunmOXZVqin5t1YWBjZW1sa2xlZGhmZ
      ]]>
    </encode>
    <decode>
      <![CDATA[
      84995052.mp3?xcode=11ed1d53d0751a9fb8f51a7d2a1689bd20b9b4351cb7f2e2&mid=0.01864442521674
      ]]>
    </decode>
    <type>8</type>
    <lrcid>0</lrcid>
    <flag>0</flag>
  </url>
  <durl>
    <encode>
      <![CDATA[
      http://zhangmenshiting2.baidu.com/data2/music/85000719/aWdjZGVtaHFfn6NndK6ap5WXcGVmm5tplWdmmGVtbGmSa5lsmpxvcWdla5ptZmtwl5doZm2YnZ1qa2ual2hwmpaTl1qin5t1YWBjZW1sa2xlZGhm
      ]]>
    </encode>
    <decode>
      <![CDATA[
      85000719.mp3?xcode=11ed1d53d0751a9f8ef89638f8048fe528bfe998fb29bead&mid=0.01864442521674
      ]]>
    </decode>
    <type>8</type>
    <lrcid>0</lrcid>
    <flag>0</flag>
  </durl>
  <url>
    <encode>
      <![CDATA[
      http://music.huoxing.com/upload/20120816/YmVnaWZob2hqYmlobJVscWhmYaGlaXZoX2tnaGVpaW1mamVnbm9wbmUy
      ]]>
    </encode>
    <decode>
      <![CDATA[ 1345128090647_5974.mp3?0.9440325582399964 ]]>
    </decode>
    <type>8</type>
    <lrcid>0</lrcid>
    <flag>0</flag>
  </url>
  <durl/>
  <url>
    <encode>
      <![CDATA[ http://zj189.cn/zj/download/music/q6qgrps2.mp3 ]]>
    </encode>
    <decode>
      <![CDATA[ zxmzf.mp3 ]]>
    </decode>
```

```
    <type>1</type>
    <lrcid>0</lrcid>
    <flag>0</flag>
  </url>
  <durl/>
  <url>
    <encode>
      <![CDATA[ http://www.dy.com.cn/studies/02/q6qgrps2.mp3 ]]>
    </encode>
    <decode>
      <![CDATA[ zxmzf.mp3 ]]>
    </decode>
    <type>1</type>
    <lrcid>0</lrcid>
    <flag>0</flag>
  </url>
  <durl/>
  <url>
    <encode>
      <![CDATA[
      http://Files.teacher.com.cn/guangdong2012/Y2JkZmVtZ2ljYmZqa WZuamRoCiIh-~opCBfqGzU$.mp3
      ]]>
    </encode>
    <decode>
      <![CDATA[ 201207012036407236最炫民族风.mp3 ]]>
    </decode>
    <type>1</type>
    <lrcid>0</lrcid>
    <flag>0</flag>
  </url>
  <durl/>
  <p2p>
    <hash>9e416694e8f16ef2fd397e0a230b0e685031439e</hash>
    <url>
      <![CDATA[ ]]>
    </url>
    <type>mp3</type>
    <size>6889567</size>
    <bitrate>192</bitrate>
  </p2p>
</result>
```

上述结果中，我们可以看到第一个url节点和第一个durl节点下的音乐链接来源都是百度，我们将只获取百度的链接。

在这里，encode节点中的前段部分和decode节点中的前段部分组成了最终的音乐链接。具体介绍如下。

在url节点下，encode下的http://zhangmenshiting.baidu.com/data2/music/84995052/和decode下的84995052.mp3?xcode=11ed1d53d0751a9fb8f51a7d2a1689bd20b9b4351cb7f2e2组成一个音乐链接，地址如下：

```
http://zhangmenshiting.baidu.com/data2/music/84995052/84995052.mp3?xcode=11ed1d53d0751a9fb8f51a7d2a1689bd20b9b4351cb7f2e2
```

在durl节点下，encode下的http://zhangmenshiting2.baidu.com/data2/music/85000719/和decode下的85000719.mp3?xcode=11ed1d53d0751a9f8ef89638f8048fe528bfe998fb29bead组成一个音乐链接，地址如下：

```
http://zhangmenshiting2.baidu.com/data2/music/85000719/85000719.mp3?xcode=11ed1d53d0751a9f8ef89638f8048fe528bfe998fb29bead
```

其中，url节点下的音乐链接是低比特率的，而durl下的链接是高比特率的，可以分别设定为音乐消息的MusicURL和HQMusicUrl参数。

8.6.2 开发实现

根据上一节的分析，在线点歌的实现代码如下：

```
 1 function getMusicInfo($entity)
 2 {
 3     if ($entity == ""){
 4         $content = "你还没告诉我音乐名称呢？";
 5     }
 6     else{
 7         if (strpos($entity, "@")){
 8             $music = explode("@",$entity);
 9             $url = "http://box.zhangmen.baidu.com/x?op=12&count=1&title=" .$music[1]."$$".$music[0]."$$$$";
10         }else{
11             $url = "http://box.zhangmen.baidu.com/x?op=12&count=1&title= ".$entity."$$";
12         }
13 
14         $ch = curl_init();
15         curl_setopt($ch, CURLOPT_URL, $url);
16         curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
17         $data = curl_exec($ch);
18 
19         $content = "检索失败";
20         try{
21             @$menus = simplexml_load_string($data, 'SimpleXMLElement', LIBXML_NOCDATA);
22             if ($menus->count > 0 && isset($menus->url[0]) && isset($menus->durl[0])){
23                 $url_prefix = substr($menus->url[0]->encode,0,strripos ($menus->url[0]->encode,'/') + 1);
24                 $url_suffix = substr($menus->url[0]->decode,0,strripos ($menus->url[0]->decode,'&'));
25                 $durl_prefix = substr($menus->durl[0]->encode,0,strripos ($menus->durl[0]->encode,'/') + 1);
26                 $durl_suffix = substr($menus->durl[0]->decode,0,strripos ($menus->durl[0]->decode,'&'));
27                 if (strpos($entity, "@")){
28                     $content = array("Title"=>$music[1],
29                                 "Description"=>$music[0],
30                                 "MusicUrl"=>$url_prefix.$url_suffix,
31                                 "HQMusicUrl"=>$durl_prefix.$durl_suffix);
32                 }else{
33                     $content = array("Title"=>$entity,
34                                 "Description"=>"百度音乐掌门人提供",
35                                 "MusicUrl"=>$url_prefix.$url_suffix,
36                                 "HQMusicUrl"=>$durl_prefix.$durl_suffix);
37                 }
38             }
39         }catch(Exception $e){
40         }
41     }
42     return $content;
43 }
```

上述代码解读如下：

第3行~第5行：判断传入内容是否为空。

第7行~第12行：根据内容中是否有“@”分隔，将前后部分分为歌手名称和歌曲名称，构造不同的查询URL。

第14行~第17行：使用cURL获取查询结果。

第21行：将结果解析成XML对象。

第22行：判断有结果返回，并且至少有一个url节点和一个durl节点。

第23行~第26行：取出url节点和durl节点中encode和decode的值。其中encode的值从开头截取到最后一个“/”，decode中的值从开头截取到最后一个“&”。

第27行~第37行：构造成音乐消息的数组。

第42行：返回音乐消息。

最后，在线点歌的运行效果如图8-9所示。

0:27
返回
微信公众平台测试...
凤凰传奇@最炫民族风
最炫民族风
凤凰传奇
最炫民族风
最炫民族风
百度音乐掌门人提供
发送

图8-9　在线点歌

8.7 一站到底

《一站到底》是江苏卫视推出的全新益智答题类节目，该档节目是由美国NBC《Who’s still standing》改编而来。这是一档形式新颖，趣味十足，在挑战中挖掘个性，又充满悬念的全新益智攻擂节目。我们在微信中也能实现类似的功能。

8.7.1 实现方案

首先，我们需要有个题库，用户答题的内容均从题库中获取。题库的结构如图8-10所示。

	字段	类型	整理	属性	空	默认	额外
☐	id	int(5)			否	无	AUTO_INCREMENT
☐	question	varchar(255)	utf8_general_ci		是	NULL	
☐	optionA	varchar(60)	utf8_general_ci		是	NULL	
☐	optionB	varchar(60)	utf8_general_ci		是	NULL	
☐	optionC	varchar(60)	utf8_general_ci		是	NULL	
☐	optionD	varchar(60)	utf8_general_ci		是	NULL	
☐	optionE	varchar(60)	utf8_general_ci		是	NULL	
☐	optionF	varchar(60)	utf8_general_ci		是	NULL	
☐	answer	varchar(6)	utf8_general_ci		是	NULL	

图8-10　一站到底题库结构

该题库的建表SQL语句如下所示：

```
DROPTABLEIFEXISTS 'questions';
CREATETABLEIFNOTEXISTS 'questions' (
  'id' int(5) NOTNULL AUTO_INCREMENT,
  'question' varchar(255) DEFAULTNULL,
  'optionA' varchar(60) DEFAULTNULL,
  'optionB' varchar(60) DEFAULTNULL,
  'optionC' varchar(60) DEFAULTNULL,
  'optionD' varchar(60) DEFAULTNULL,
  'optionE' varchar(60) DEFAULTNULL,
  'optionF' varchar(60) DEFAULTNULL,
  'answer' varchar(6) DEFAULTNULL,
PRIMARYKEY ('id')
) ENGINE=MyISAM  DEFAULT CHARSET=utf8;
```

上述结构中，id表示题目的序号，为表中的主键，question为题目的内容，optionA~optionF表示该题可选择的答案有A到F6个选项，但不是每个题目都一定有这么多选项，大部分选择题都为4个选项，所以允许答案选项为空，answer字段为该题的正确答案。

有了表以后，需要往表中添加题库内容，一个上传好的题库内容如图8-11所示。

	question	optionA	optionB	optionC	optionD	optionE	optionF	answer
1	属于国家一类保护动物的是：	天鹅	娃娃鱼	孔雀	丹顶鹤	<NULL>	<NULL>	A
2	举重时运动员为什么搓白粉？	使手变的粗	吸取手上的	刺激肌肉	保护皮肤	<NULL>	<NULL>	B
3	非洲国家边界按什么划分的最	直线或曲线	经线或纬线	河流、山脉	民族	<NULL>	<NULL>	B
4	举重比赛的级别是按：	按杠铃重量	按体重划分	按年龄划分	按性别划分	<NULL>	<NULL>	B
5	问仓储超市与其它超市最大的不	铺房简陋	储售场地合	随进随卖	需要库房	<NULL>	<NULL>	B
6	18世纪德国贵族热衷于收集并关	书籍	与情人来往	首饰	树叶、果实	日记	金币	D
7	鸵鸟受到惊吓或敌害时可能会：	把头埋进沙	将脖子平贴	把全身都埋	会将头低垂	<NULL>	<NULL>	D
8	最早制造香水瓶所用的原料是：	陶瓷	玻璃	石膏	铜瓶	<NULL>	<NULL>	C
9	持有哪种"伤残证书"，可在乘坐	革命伤残军	公安人员残	革命工作人	参战民兵残	参战民工残	<NULL>	A
10	最早的四冲程内燃机用的是什么	酒精	煤气	汽油	柴油	<NULL>	<NULL>	B
11	下面哪一种熨烫衣服的方法是不	熨之前在衣	尽可能长时	掌握合适的	在衣服和熨	<NULL>	<NULL>	B
12	冻疮是由于皮肤哪一部分受损而	毛囊	汗腺	竖毛肌	毛细血管	<NULL>	<NULL>	D
13	世界上最大的宫殿是：	故宫	白宫	爱丽舍宫	克里姆林宫	<NULL>	<NULL>	A
14	我国曲艺中的相声是：	北方曲种	南方曲种	<NULL>	<NULL>	<NULL>	<NULL>	A
15	世界上最靠北的首都是：	斯德哥尔摩	莫斯科	雷克亚未克	哥本哈根	<NULL>	<NULL>	C
16	反映波兰作曲家肖邦生平的美国	《翠堤春晓	《一曲难忘	《音乐之声	《雨中曲》	<NULL>	<NULL>	B
17	除法律另有规定或合同另有约定	投保人	保险人	被保险人	受益人	<NULL>	<NULL>	A
18	国际奥委会为感谢萨马兰奇对奥	邮票	邮折	纪念封	纪念币	<NULL>	<NULL>	C

图8-11　一站到底题库内容

然后，需要有个保存用户答题记录的表，以便为用户的答题情况记录分数等情况。记录表的结构如图8-12所示。

	字段	类型	整理	属性	空	默认
☐	opcnid	varchar(28)	gbk_chinese_ci		否	无
☐	question	int(5)			是	-1
☐	question_y	int(5)			是	0
☐	question_n	int(5)			是	0

图8-12　一站到底用户记录表结构

该记录表的建表SQL语句如下所示：

```
DROPTABLEIFEXISTS 'record';
CREATETABLEIFNOTEXISTS 'record' (
  'openid' varchar(28) NOTNULL COMMENT '微信ID',
  'question' int(5) DEFAULT'-1' COMMENT '答题偏移',
  'question_y' int(5) DEFAULT'0' COMMENT '答题正确',
  'question_n' int(5) DEFAULT'0' COMMENT '答题错误',
PRIMARYKEY ('openid')
) ENGINE=MyISAM DEFAULT CHARSET=gbk;
```

上述结构中，openid表示微信用户的id号，为表中的主键；question为答题偏移，表示用户当前答题数量，我们设置默认值为-1，当用户开始参与答题后，程序中将设置该值为0；question_y表示该用户当前答题正确数目，默认值为0；question_y表示该用户当前答题错误数目，默认值为0。

一站到底在微信中的实现流程如下：

1）用户回复“答题”。

2）系统返回题目的内容及选项。

3）用户发送自己选择的答案。

4）系统判断答案是否正确。

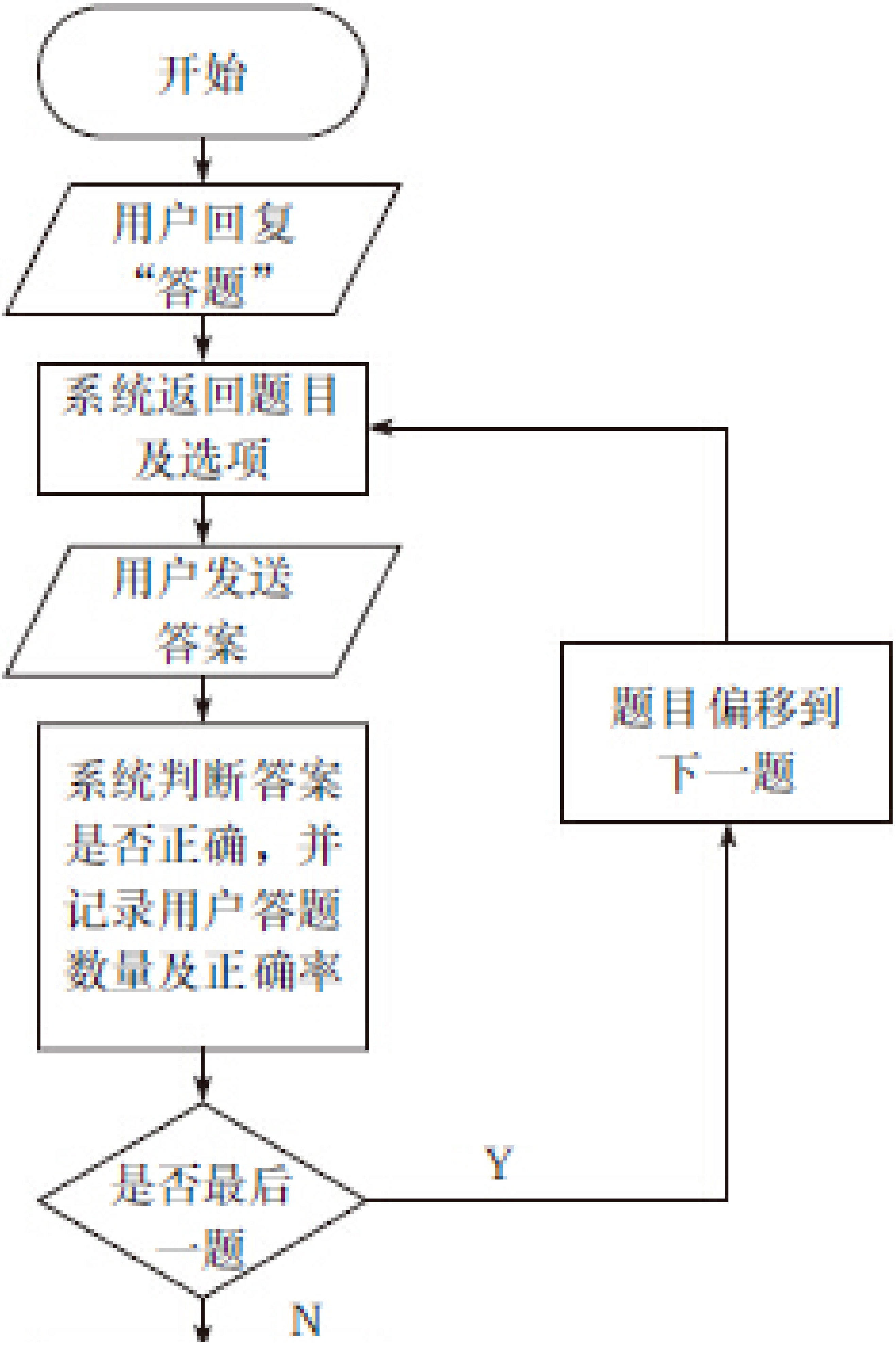
5）记录用户答题情况及进度。

6）系统回复下一题内容及选项。

7）重复步骤直到最后一题。

8）系统回复用户答题情况。

一站到底整个系统的流程图如图8-13所示。



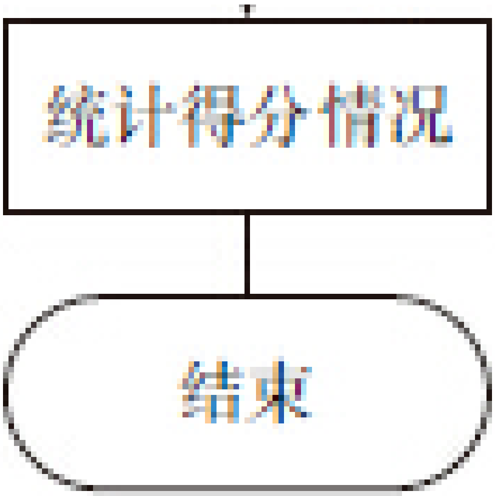


图8-13　一站到底流程图

8.7.2　开发实现

首先，当我们接收到用户发过来的文字消息的时候，判断是否是答题有关的输入，这里使用“答题”作为进入答题开始的指令，使用A到F共6个字母（不区分大小写）作为答案的输入指令。相关代码如下：

```
private function receiveText($object)
{
    $keyword = trim($object->Content);
    if ($keyword == "答题" || preg_match("/^[A-Fa-f]$/", $keyword)){
        include("question.php");
        $content = getQuestionInfo($object->FromUserName, (($keyword == "答题")?$keyword:strtoupper(trim($object->Content))));
    }
    $result = $this->transmitText($object, $content);
    return $result;
}
```

上述代码中，判断用户输入是否为“答题”，并使用正则表达式判断字母范围。如果符合，则将用户的id以及答题命令传给函数getQuestionInfo（），在这个过程中，将字母强制转换成了小写。

答题部分的程序实现如下：

```
function getQuestionInfo($openid, $answer)
{
    if(isset($_SERVER['HTTP_APPNAME'])){        //SAE
        $mysql_host = SAE_MYSQL_HOST_M;
        $mysql_host_s = SAE_MYSQL_HOST_S;
        $mysql_port = SAE_MYSQL_PORT;
        $mysql_user = SAE_MYSQL_USER;
        $mysql_password = SAE_MYSQL_PASS;
        $mysql_database = SAE_MYSQL_DB;
    }else{
        $mysql_host = "127.0.0.1";
        $mysql_host_s = "127.0.0.1";
        $mysql_port = "3306";
        $mysql_user = "root";
        $mysql_password = "root";
        $mysql_database = "weixin";
    }

    $con = mysql_connect($mysql_host.':'.$mysql_port, $mysql_user, $mysql_password);
    if (!$con){die('Could not connect: ' . mysql_error());}
    mysql_query("SET NAMES 'UTF8'");
    mysql_select_db($mysql_database, $con);

    $mysql_table_questions = "questions";
    $mysql_get_rows = "SHOW TABLE STATUS like '".$mysql_table_questions."';";
    $result5 = mysql_query($mysql_get_rows);
    $row = mysql_fetch_array($result5);
    $max_question = $row['Rows'];

    $mysql_table_record = "record";
    $finish_question = 0;
    if ($answer == "答题"){
        $mysql_insert = "INSERT INTO `".$mysql_table_record."` (`openid`, `question`, `question_y`, `question_n`) VALUES('".$openid."', '0', '0', '0') ON DUPLICATE KEY UPDA
        @mysql_query($mysql_insert);
        $finish_question = 0;
    }else{
        $mysql_select = "SELECT * FROM `".$mysql_table_record."
` WHERE `openid` = '".$openid."';";
        $result4 = mysql_query($mysql_select);
        $row = mysql_fetch_array($result4);
        $finish_question = $row['question'];

        if($finish_question == $max_question){
            return "你已经答完了全部题目";
        }

    $mysql_update = "UPDATE `".$mysql_table_record."` SET `question` = `question` + 1 WHERE `openid` = '".$openid."';";
        $result3 = mysql_query($mysql_update);
        $finish_question += 1;
    }

    if ($finish_question == 0){
    $mysql_state = "SELECT * FROM `".$mysql_table_questions."
` LIMIT ".$finish_question.",1";
    }else if($finish_question == $max_question){
    $mysql_state = "SELECT * FROM `".$mysql_table_questions."
` LIMIT ".($finish_question - 1).",1";
    }else{
    $mysql_state = "SELECT * FROM `".$mysql_table_questions."` LIMIT ". ($finish_question - 1).",2";
    }
    $result = @mysql_query($mysql_state);

    $content = "";
    if ($finish_question == 0){
        if (mysql_num_rows($result) < 1){
            $content .= "系统繁忙，请过会重试\n";
        }else{
```

```
 65             $content .= "一站到底在线答题开始\n";
 66             while($row = mysql_fetch_array($result))
 67             {
 68                 $content .= "第".$row['id']."题: ".$row['question']. "\n";
 69                 $content .= empty($row['optionA'])?"":"A. ".$row ['optionA']. "\n";
 70                 $content .= empty($row['optionB'])?"":"B. ".$row ['optionB']."\n";
 71                 $content .= empty($row['optionC'])?"":"C. ".$row ['optionC']."\n";
 72                 $content .= empty($row['optionD'])?"":"D. ".$row ['optionD']."\n";
 73                 $content .= empty($row['optionE'])?"":"E. ".$row ['optionE']."\n";
 74                 $content .= empty($row['optionF'])?"":"F. ".$row ['optionF']."\n";
 75             }
 76           }
 77     }else if($finish_question == $max_question){
 78        while($row = mysql_fetch_array($result))
 79        {
 80             $content .= ($answer == $row['answer'])?"回答正确\n":"回答错误! 正确答案为 ".$row['answer']."\n";
 81             if ($answer == $row['answer']){
 82                 $mysql_update_yesno = "UPDATE `".$mysql_table_record."` SET `question_y` = `question_y` + 1 WHERE `openid` = '".$openid."';";
 83              }else{
 84                 $mysql_update_yesno = "UPDATE `".$mysql_table_record."` SET `question_n` = `question_n` + 1 WHERE `openid` = '".$openid."';";
 85              }
 86              $result_yesno = mysql_query($mysql_update_yesno);
 87
 88              $mysql_statistics = "SELECT * FROM `".$mysql_table_record."` WHERE `openid` = '".$openid."';";
 89              $result6 = mysql_query($mysql_statistics);
 90              $row = mysql_fetch_array($result6);
 91              $result_all = $row['question'];
 92              $result_yes = $row['question_y'];
 93              $content .= "恭喜, 您已完成全部题目。"."\n题目个数: ".$result_all."\n正确个数: ".$result_yes."\n正确率: ".round(($result_yes / $result_all) * 100, 2)."%";
 94          }
 95     }else{
 96         $first = true;
 97         while($row = mysql_fetch_array($result))
 98         {
 99             if ($first){
100                 $content .= ($answer == $row['answer'])?"回答正确\n":"回答错误! 正确答案为 ".$row['answer']."\n";
101                 $first = false;
102                 if ($answer == $row['answer']){
103     $mysql_update_yesno = "UPDATE `".$mysql_table_record."` SET `question_y` = `question_y` + 1 WHERE `openid` = '".$openid."';";
104                 }else{
105                     $mysql_update_yesno = "UPDATE `".$mysql_table_record."` SET `question_n` = `question_n` + 1 WHERE `openid` = '".$openid."';";
106                 }
107                 $result_yesno = mysql_query($mysql_update_yesno);
108             }else{
109                $content .= "第".$row['id']."题: ".$row['question']. "\n";
110                $content .= empty($row['optionA'])?"":"A. ".$row ['optionA']."\n";
111                $content .= empty($row['optionB'])?"":"B. ".$row ['optionB']."\n";
112                $content .= empty($row['optionC'])?"":"C. ".$row ['optionC']."\n";
113                $content .= empty($row['optionD'])?"":"D. ".$row ['optionD']."\n";
114                $content .= empty($row['optionE'])?"":"E. ".$row ['optionE']."\n";
115                $content .= empty($row['optionF'])?"":"F. ".$row ['optionF']."\n";
116             }
117         }
118     }
119     mysql_close($con);
120     return trim($content);
121 }
```

上述代码解读如下:

第3行~第17行:定义数据库环境变量,此处判断程序运行环境是SAE还是本地服务器环境,分别定义不同的环境变量。

第19行~第22行:进行数据库连接,设置字符集为“UTF8”,设置活动的MySQL数据库。

第24行~第28行:查询得到当前题库中题目的数量,便于下面代码中进行判断。

第32行~第36行:当用户输入“答题”时,初始化用户答题记录,如果已经有记录,则清零,并且设置当前用户答题数量为0。

第37行~第40行:如果用户输入的是答案的字母,则先查找到用户当前的答题记录个数。

第42行~第44行:如果用户的答题个数已经为题目数量,那么说明用户已经完成答题,不需要再参加此次答题了。

第46行~第48行:如果还不等于题目数量,那么说明没有答完,这里将用户的答题数增加1。

第51行~第57行:根据用户答题个数,定义不同的查询语句。当用户答第一题时,取回第一题的题目。当用户答最后一题的时候,需要取最后一题的答案,而在其他情况下,需要取回当前题目的答案及下一题的题目。这是SQL语句中LIMIT子句设置不同的返回数量的原因。

第58行:执行上述查询语句。

第61行~第76行:这段代码是获取当用户题数为0时的返回内容,表示用户没有参加过答题,首先加上一段一站到底的提示语,再将第一题的题目及选项返回给用户。如果没有查询到题目,则可能是系统出现异常。

第77行~第94行:这段代码是处理用户答到最后一题时的情况。这时只需要判断用户答案是否正确,然后将用户的答题情况记录到表中,最后,再统计一下用户的答题情况,包括答题个数、正确个数及正确率。

第96行~第118行:这是处理既不是第一题,也不是最后一题的情况。这种情况下,SQL查询语句中会返回两条记录,第一条是当前题目,第二条是下一个题目。系统获取当前题目的答案用于判断用户是否回答正确,获取下一题的内容及选项返回给用户,以便用户回答下一题。

第119行:关闭数据库连接。

第120行:返回内容。

一站到底的运行效果如图8-14所示。

图8-14　一站到底

8.8　智能聊天机器人

小i机器人是上海智臻网络科技有限公司开发的智能机器人产品，是全球用户量最大的中文聊天机器人。它最早在MSN上出现，后拓展到QQ、短信、网页、微博、微信等交互平台上，第一个手机版本发布于2012年2月，总用户已超过1亿。该机器人能查询新闻、地图、餐饮、机票等各类生活资讯。

8.8.1　iBotCloud

iBotCloud全称为“小i机器人智能云服务平台”，是“小i机器人”在开放自身的核心能力（包括语义理解、智能问答、智能语音、知识管理和交互渠道接入等）的基础上，为广大开发者、合作伙伴和企业提供的一系列基于云计算和大数据平台的在线服务和开发接口。

开发者可以利用iBotCloud提供的在线服务，定义机器人属性、编辑机器人的智能问答，打造属于您的智能交互机器人。此外，iBotCloud支持多种渠道（微信、腾讯微博、新浪微博、iPhone SDK、Android SDK）的快速接入。

iBotCloud的官方网址如下：

```
http://cloud.xiaoi.com/index.jsp
```

用户通过注册即可获得“初级测试权限”，获得智能问答、自定义知识模块的基础测试权限；初级授权用户通过提交个人 / 企业真实信息可以向平台申请“高级测试权限”，审核通过后即可拥有智能问答、自定义知识、语音识别、语音合成板块的高级测试权限。获得高级测试权限后，每月有10000次智能问答的授权调用。图8-15是高级测试授权功能列表图。

账号权限：高级测试授权 ? 查看授权等级说明

功能授权

功能	当前授权 ?	剩余次数 ?	过期日期
智能问答 ?	10,000次/月	9,967次	2014-05-28
语音识别 ?	50次/日	50次	2014-05-28
语音合成 ?	50次/日	50次	2014-05-28

快速接入一览

渠道名称	微信	易信	腾讯微博
授权状态 ?	无限制	无限制	无限制
关联账号 ?	0	0	0

图8-15　高级测试授权功能列表

同时，iBotCloud会为开发者提供API接口（Key和Secret），开发者可以通过访问API接口与iBotCloud进行快速对接，从而为自己的应用程序增加丰富的智能语音交互体验。

iBotCloud智能问答接口地址如下：

```
http://nlp.xiaoi.com/robot/ask.do
```

该接口的相关参数如表8-8所示。

表8-8　智能问答接口参数说明

参数	示例	说明
userId	user0001	用户 id
question	您好	问题内容
type	1	响应格式（0- 基础、1- 高级）
platform	weixin	平台（weixin- 微信、yixin- 易信、tqq- 腾讯微博、custom- 自定义）

响应格式为普通响应的参数说明如表8-9所示。

表8-9　普通响应格式参数说明

参数	说明	示例
无	直接纯文本响应	为纯文本类型，例如：“你好，我的名字叫小 i”

该接口请求时的原始数据如下：

```
POST http://nlp.xiaoi.com/ask.do?platform=weixin HTTP/1.1
Host: nlp.xiaoi.com
Connection: keep-alive
Content-Length: 70
Origin: http://nlp.xiaoi.com
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 5.1) AppleWebKit/537.1 (KHTML, like Gecko) Chrome/21.0.1180.89 Safari/537.1
Content-Type: application/x-www-form-urlencoded
Accept: text/plain, */*; q=0.01
X-Auth: app_key="zIp6yeleNdHp", nonce="a2a303c2963ea105823b5d798ab2d96302f3", signature="a7a9617d8401557776e99539bc608fa85bf377bd"
Referer: http://nlp.xiaoi.com/invoke.html?ts=1393684911322
Accept-Encoding: gzip,deflate,sdch
Accept-Language: en-US,en;q=0.8
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3
Cookie: _ga=GA1.2.563236203.1393475725

question=%E4%BD%A0%E5%A5%BD&userId=o7Lp5t6n59DeX3U0C7Kric9qEx-Q&type=0
```

请求后得到的响应数据如下所示：

```
请求后得到的响应数据如下所示：
HTTP/1.1 200 OK
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Cache-Control: no-cache
Pragma: no-cache
Content-Type: text/plain; charset=utf-8
Content-Length: 49

你好,我是小方,很高兴认识你。
```

8.8.2 开发实现

微信公众账号在收到文本消息之后，即调用小i机器人来自动回复，需要将用户id及用户的内容都传给机器人。相关代码如下所示：

```
private function receiveText($object)
{
    $keyword = trim($object->Content);
    include("xiaoi.php");
    $content = getXiaoiInfo($object->FromUserName, $keyword);
    $result = $this->transmitText($object, $content);
    return $result;
}
```

小i机器人的逻辑实现比较复杂，其智能问答的实现代码如下：

```
1 function getXiaoiInfo($openid, $content)
2 {
3     //定义app
4     $app_key="************";
5     $app_secret="********************";
6
7     //签名算法
8     $realm = "xiaoi.com";
9     $method = "POST";
10    $uri = "/robot/ask.do";
11    $nonce = "";
12    $chars = "abcdefghijklmnopqrstuvwxyz0123456789";
13    for ($i = 0; $i < 40; $i++) {
14        $nonce .= $chars[ mt_rand(0, strlen($chars) - 1) ];
15    }
16    $HA1 = sha1($app_key.":".$realm.":".$app_secret);
17    $HA2 = sha1($method.":".$uri);
18    $sign = sha1($HA1.":".$nonce.":".$HA2);
19
20    //接口调用
21    $url = "http://nlp.xiaoi.com/robot/ask.do";
22    $ch = curl_init();
23    curl_setopt($ch, CURLOPT_URL, $url);
24     curl_setopt($ch, CURLOPT_HTTPHEADER, array('X-Auth: app_key="'.$app_key.'", nonce="'.$nonce.'", signature="'.$sign.'"'));
25    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
26    curl_setopt($ch, CURLOPT_POST, 1);
27     curl_setopt($ch, CURLOPT_POSTFIELDS, "question=".urlencode($content)."&userId=".$openid."&platform=custom&type=0");
28    $output = curl_exec($ch);
29    if ($output === FALSE){
30        return "cURL Error: ". curl_error($ch);
31    }
32    return trim($output);
33 }
```

上述代码解读如下：

第3行~第5行：定义API接口，即申请到的接口。

第7行~第18行：这一段为签名算法的实现。小i机器人API需要通过签名来访问，签名的过程是将APP的key和secret以及随机数等参数根据一定签名算法生成的签名值作为新的请求头中的一部分，以此来提高访问过程中的防篡改性。

签名算法如下：

1）sha1加密（app_key：realm：app_secret）其中realm为“xiaoi.com”。

2）sha1加密（method：uri）其中method为请求方法，如“POST”，uri为“/robot/ask.do”。

3）sha1加密（HA1：nonce：HA2）其中HA1为步骤1的值，HA2为步骤2的值，nonce为40位随机数。

第20行~第31行：使用cURL获取调用智能问答的结果。

小i机器人API的有效访问都必须包含签名请求头，在第24行中定义了这样一个签名头字符串，其中app_key为API接口的key，nonce为上面过程中生成的40位随机数，signature为签名算法步骤3的值。最后为该字符串添加请求头：“X-Auth”。

第27行：将请求参数连接成字符串，通过POST提交给接口。

第32行：返回接口的内容。

小i机器的运行效果如图8-16所示。

23:30
返回
方倍工作室
你好
你好，我是小方，很高兴认识你。
FromUserName又称什么?
你再说这些字典里也没有的东西，我就关门放旺财啦！
哈哈
发送

图8-16　小i机器人智能聊天

8.9　本章小结

本章介绍了笑话、星座运势、周公解梦、姓名测试、人脸识别夫妻相、在线点歌、一站到底答题、智能聊天机器人等功能的实现，这些功能能够为公众账号增加娱乐功能，帮助运营者提高公众账号的价值。

第9章　企业类应用开发

越来越多的企业开始申请微信公众平台，微信公众平台成为企业在移动互联网上进行品牌传播、吸引用户、商家促销等的重要战略渠道。本章将介绍一些企业微信公众平台上常用的功能开发，这些功能包括：会员卡、预约订单、产品相册、大转盘、地图导航、智能回复、客服在线。企业级功能的开发往往比较复杂，对功能及效果的要求比较高，我们可以借鉴已有资源来修改、完善这些功能。

9.1　会员卡

会员卡一般是指企业发行的用于识别顾客身份的卡片，普遍用于服务行业。它是公司形象宣传的理想载体，能起到吸引新顾客、留住老顾客、增强顾客忠诚度的作用，还能实现打折、积分等功能。本节介绍如何在微信公众平台上实现会员卡的功能。

9.1.1　页面布局

会员卡开发首先需要准备素材，这里主要包括会员卡的正面及反面图片——大小固定的透明企业Logo以及页面背景图案。图9-1显示了会员卡正反面图片。

图9-1　会员卡素材

接着制作HTML页面，其中body中的代码如下：

```
 1 <div id="mappContainer">
 2   <section id="card_ctn">
 3     <div class="bk1"></div>
 4     <div class="cont">
 5       <div class="card">
 6         <div class="front">
 7           <figure class="fg">
 8             <img src="./companylogo.png">
 9             <figcaption class="fc">
10               <span class="cname">微信会员卡</span>
11               <span class="t"></span>
12               <span class="n">NO.80000716</span>
13             </figcaption>
14           </figure>
15         </div>
16         <div class="back">
17           <figure class="fg">
18             <div class="info">
19               <p class="addr">广东省深圳市南山区滨海大道深圳湾体育中心</p>
20               <p class="tel"><a class="autotel" href="tel:07558620000">0755-
86280000</a></p>
21             </div>
22             <p class="keywords">番茄田</p>
23           </figure>
24         </div>
25       </div>
26     </div>
27   </section>
28
29   <div id="vip">
30     <small><em>尊贵会员，尽享如下特权:</em></small>
31     <ul class="round">
32       <li data-ajax-params="" data-ajax-act="" class="only"><a href=""> <b>9折优惠券</b></a></li>
33     </ul>
34   </div>
35 </div>
```

上述代码解读如下：

第3行：定义bk1的类，用于显示背景图案。

第5行~第16行：定义会员卡正面信息，主要包括会员卡正面图片、公司Logo图案、“微信会员卡”字样、会员卡编号。

第16行~第24行：定义会员卡反面信息，主要包括会员卡反面图片、公司信息、公司地址、电话等。

使用CSS来控制各元素的层叠方式，相关代码如下：

```
.fg{margin:0 auto;position:relative;width:236px;height:143px;-webkit-background-size:236px 143px;background-size:236px 143px;background-position:0 0;background-repeat:no-repe
.fg.barcode{text-align:left;}
.fg img:nth-child(1){margin-top:25px;max-width:185px;}
.fg.barcode img:nth-child(1){margin:10px 0 5px 10px;max-height:40px;}
.fg img:nth-child(2){width:234px;margin-left:1px;}
.fg .info{height:83px;padding-top:28px;}
.fg .info .addr,.fg .info .tel{display:block;text-align:left;color:#666;font-size:14px;margin-left:12px;padding:0 20px 12px 23px;line-height:18px;background-position:0 2px;ba
.fg .info .addr{background-image:url('http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../img/icon_addr.png');}
.fg .info .tel{background-image:url('http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../img/icon_tel.png');}
.fg .keywords{height:25px;line-height:25px;text-align:center;color:#fff;font-size:10px;}
.fc{width:210px;height:125px;line-height:20px;padding:8px 7px 0 9px;overflow:hidden;position:absolute;top:0;left:4px;border-radius:0 0 4px 4px;font-size:13px;text-align:left;
.fc .cname,.fc .t,.fc .n{display:inline-block;height:20px;overflow:hidden;white-space:nowrap;}
.fc .cname{width:222px;height:16px;margin:0 0 89px 0;line-height:16px;font-size:10px;}
.fc .t{width:106px;font-size:12px;}
.fc .n{width:82px;float:right;text-align:right;font-size:10px;}
```

再使用CSS3的3D转换功能，对会员卡相关元素重新格式化，实现会员卡正反面翻转的效果，CSS中定义代码如下。

```
.card .front,.card .back{position:absolute;top:0;left:0;-webkit-transform-style:preserve-3d;-moz-transform-style:preserve-3d;-ms-transform-style:preserve-3d;transform-style:pr
.card .front{-webkit-transform:rotateY(0deg);-moz-transform:rotateY(0deg);-ms-transform:rotateY(0deg);transform:rotateY(0deg);z-index:1;}
.card.flip .front{-webkit-transform:rotateY(180deg);-moz-transform:rotateY(180deg);-ms-transform:rotateY(180deg);transform:rotateY(180deg);z-index:0;}
.card .back{-webkit-transform:rotateY(-180deg);-moz-transform:rotateY(-180deg);-ms-transform:rotateY(-180deg);transform:rotateY(-180deg);z-index:0;}
.card.flip .back{-webkit-transform:rotateY(0deg);-moz-transform:rotateY(0deg);-ms-transform:rotateY(0deg);transform:rotateY(0deg);z-index:1;}
```

9.1.2　开发实现

为了存储用户信息，需要创建会员卡表“member”，SQL语句如下：

```
DROPTABLEIFEXISTS 'member';
CREATETABLEIFNOTEXISTS 'member' (
  'id' int(11) NOTNULL AUTO_INCREMENT,
  'openid' varchar(28) NOTNULL COMMENT '微信ID',
  'cardid' varchar(8) DEFAULTNULL COMMENT '卡号',
PRIMARYKEY ('id')
) ENGINE=MyISAM  DEFAULT CHARSET=gbk AUTO_INCREMENT=7 ;
--
-- 表中的数据 'member'
--
INSERTINTO 'member' ('id', 'openid', 'cardid') VALUES
(1, 'oDeOAjj54GvEkkgCV2d7QV4-JLMc', '80000718');
```

在该表中定义了3个字段，id、openid、cardid分别表示序号、用户的微信id以及生成的卡号，在该表中，已经插入一条测试数据。我们可以在用户关注或者查看会员卡的时候，根据一定的规则自动生成该用户的会员卡记录。

当用户发送“会员”或者“会员卡”时，将回复一条图文消息，该图文消息的链接为会员卡的页面并带上会员的微信id作为参数，核心代码如下所示：

```
private function receiveText($object)
{
    $keyword = trim($object->Content);
    $category = substr($keyword,0,6);
    $code = trim(substr($keyword,6,strlen($keyword)));
    switch ($category)
    {
        case "会员":
            $content[] = array("Title" =>"会员卡",
            "Description" =>"点击图片领取会员卡",
            "PicUrl" =>"http://pic2.ooopic.com/01/38/77/75bOOOPICea.jpg",
            "Url" =>"http://cctv15.sinaapp.com/member.php?openid=".$object->FromUserName);
```

```
            break;
        default:
            $content = "";
            break;
    }
    if(is_array($content)){
        $result = $this->transmitNews($object, $content);
    }else{
        $result = $this->transmitText($object, $content);
    }
    return $result;
}
```

用户发送“会员”时的效果如图9-2所示。

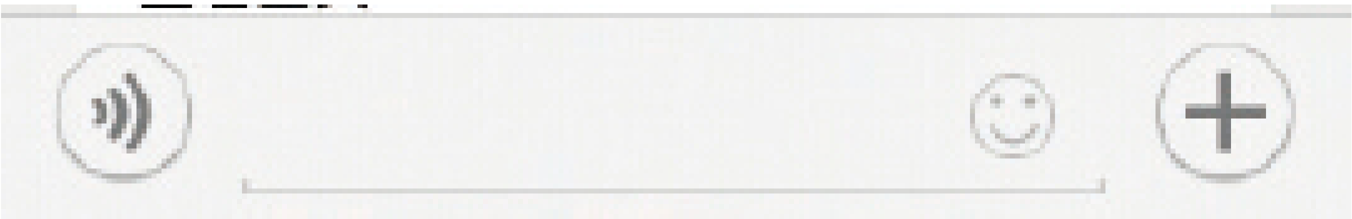


图9-2　会员卡图文消息

单击图文消息后，用户将访问类似如下地址：

```
http://cctv15.sinaapp.com/member.php?openid=oDeOAjj54GvEkkgCV2d7QV4-JLMc
```

“member.php”文件将通过$_GET变量来获取表单中openid的数据，再通过openid从数据库的member表中查询到用户的会员卡卡号cardid。相关代码如下：

```
$openid = $_GET["openid"];
$cardid = fromOpenID2CardID($openid);

function fromOpenID2CardID($openid)
{
    $mysql_host = SAE_MYSQL_HOST_M;
    $mysql_host_s = SAE_MYSQL_HOST_S;
    $mysql_port = SAE_MYSQL_PORT;
    $mysql_user = SAE_MYSQL_USER;
    $mysql_password = SAE_MYSQL_PASS;
    $mysql_database = SAE_MYSQL_DB;

    $mysql_table = "member";
    $mysql_state = "SELECT * FROM ".$mysql_table." WHERE `openid` = '".$openid."'";

    $con = mysql_connect($mysql_host.':'.$mysql_port, $mysql_user, $mysql_password, true);
    if (!$con){
        die('Could not connect: ' . mysql_error());
    }
    mysql_query("SET NAMES 'UTF8'");
    mysql_select_db($mysql_database, $con);
    $result = mysql_query($mysql_state);
    $cardID = "";
    while($row = mysql_fetch_array($result))
    {
        $cardID = $row['cardid'];
        break;
    }
    mysql_close($con);
    return $cardID;
}
```

会员卡的最终效果如图9-3所示。

图9-3　会员卡

9.2　预约订单

预约是指对某项事情提前做出安排。客户使用预约可以减少排队时间，避免时间浪费；企业在接收到顾客的预约后，可以提前做好准备，提高服务速度。本节介绍如何在微信公众台上开发预约订单功能。

9.2.1　页面布局

预约订单开发主要在于两个部分，一个是用户填写表单，另一个是信息的确认页面。

用户填写表单的页面是第一个Web页面，在该Web页面上方可以放置公司的Logo或者宣传图片，相关代码如下：

```
<div class="banner">
  <div id="wrapper">
    <div id="scroller">
      <ul id="thelist"><img src="img/logo.png" alt="预约口腔医生" style= "width:100%">
      </ul>
    </div>
  </div>
  <div class="clr"></div>
</div>
```

下方放置页面功能介绍及订单填写表单，其中页面介绍相关代码如下：

```
<ul class="round">
  <li>
    <h2>预约口腔医生</h2>
    <div class="text">
      长沙市方倍口腔竭诚为您服务<br/>
    联系电话：0731-84808888</div>
  </li>
</ul>
```

最后，需要让用户填写表单，并将表单信息通过POST提交给submit.php文件。相关代码如下：

```
<form method="post" action="submit.php" id="form" onsubmit="return tgSubmit()">
  <ul class="round">
    <li class="title mb"><span class="none">请填写以下信息</span></li>
    <li class="nob">
      <table width="100%" border="0" cellspacing="0" cellpadding="0" class="kuang">
```

```
6         <tbody>
7           <tr>
8             <th>姓名</th>
9             <td><input type="text" class="px" placeholder="请输入姓名" id="name" name="name" value="">
10            </td>
11          </tr>
12        </tbody>
13      </table>
14    </li>
15    <li class="nob">
16      <table width="100%" border="0" cellspacing="0" cellpadding="0" class="kuang">
17        <tbody>
18          <tr>
19            <th>性别</th>
20     <td><select id="sex" name= "sex" class="dropdown-select"><option value="" selected="">请选择性别
</option><option value="男">男</option><option value="女">女</option></select>
21            </td>
22          </tr>
23        </tbody>
24      </table>
25    </li>
26    <li class="nob">
27      <table width="100%" border="0" cellspacing="0" cellpadding="0" class="kuang">
28        <tbody>
29          <tr>
30            <th>年龄</th>
31            <td><input type="text" class="px" placeholder="请输入年龄" id="age" name="age" value="">
32            </td>
33          </tr>
34        </tbody>
35      </table>
36    </li>
37    <li class="nob">
38      <table width="100%" border="0" cellspacing="0" cellpadding="0" class="kuang">
39        <tbody>
40          <tr>
41            <th>手机</th>
42            <td><input type="text" class="px" placeholder="请输入手机" id="mobile" name="mobile" value="">
43            </td>
44          </tr>
45        </tbody>
46      </table>
47    </li>
48    <li class="nob">
49      <table width="100%" border="0" cellspacing="0" cellpadding="0" class="kuang">
50        <tbody>
51          <tr>
52            <th>预约日期</th>
53            <td>
54              <select id="bookdate" name= "bookdate" class="dropdown-select">
55                <option value="" selected="">请选择预约日期</option>
56                <?php
57                for ($i = 1; $i <= 6; $i++) {
58                  $offset = strtotime("+".($i-1)." day");
59                  $bDate = date("m月d日",$offset);
60                  $optionString .= '<option value="'.$bDate.'">'. $bDate.'</option>';
61                }
62                echo $optionString;
63                ?>
64              </select>
65            </td>
66          </tr>
67        </tbody>
68      </table>
69    </li>
70    <li class="nob">
71      <table width="100%" border="0" cellspacing="0" cellpadding="0" class="kuang">
72        <tbody>
73          <tr>
74            <th>预约专家</th>
75     <td><select id="bookexpert" name="bookexpert" class="dropdown-select"><option value="" selected="">请选择预约专家</option><option value="陈艳">陈艳</option><option value='
76            </td>
77          </tr>
78        </tbody>
79      </table>
80    </li>
81  </ul>
82
83  <div class="footReturn">
84     <input type="hidden" name="openid" value="<?php echo $_GET["openid"];?>">
85     <input type="submit" class="submit" value="提交信息">
86  </div>
87 </form>
```

在上述代码中，定义了一个form表单，设置了“姓名”、“性别”、“年龄”、“手机”、“预约日期”、“预约专家”等选项，其中“姓名”、“年龄”、“手机”由用户自己填写，“性别”、“预约专家”由程序指定选择列表供用户选择，“预约日期”则由PHP程序生成未来6天的日期，在提交的表单信息中还将用户的微信id隐藏起来一起提交（第84行）。

对这些表单选项还需要使用JavaScript来校验用户是否正确输入，并在错误输入的时候显示提示语，相关代码如下所示：

```
<script>
    function showTip(tipTxt) {
      var div = document.createElement('div');
      div.innerHTML = '<div class="deploy_ctype_tip"><p>' + tipTxt + '</p></div>';
      var tipNode = div.firstChild;
      $("#wrap").after(tipNode);
      setTimeout(function () {
        $(tipNode).remove();
      }, 1500);
    }
    function tgSubmit(){
      var name=$("#name").val();
      if($.trim(name) == ""){
        showTip('请输入姓名')
        return false;
      }
      var sex=$("#sex").val();
      var year=$("#year").val();
      var patrn = /^[0-9]{1,2}$/;
      if (!patrn.exec($.trim(year))) {
        showTip('请输入年龄')
        return false;
      }
      var mobile=$("#mobile").val();
      if($.trim(mobile) == ""){
        showTip('请正确填写手机号码')
        return false;
      }
      var patrn = /^13[0-9]{9}$|^15[0-9]{9}$|^18[0-9]{9}$/;
      if (!patrn.exec($.trim(mobile))) {
        showTip('请正确填写手机号码')
        return false;
      }
      var booktime=$("#booktime").val();
      if($.trim(booktime) == ""){
        showTip('请输入预约日期')
        return false;
      }
      var bookexpert=$("#bookexpert").val();
      if($.trim(bookexpert) == ""){
        showTip('请输入预约专家')
        return false;
      }
      return true;
```

```
  }
</script>
```

上述页面的实现如图9-4所示。

21:30
预约口腔医生
预约口腔医生
长沙市方倍口腔竭诚为您服务
联系电话：0731-1234567
请填写以下信息
姓名
请输入姓名

图9-4　订单预约页面

9.2.2　开发实现

当用户单击“提交信息”按钮之后，这些表单信息就被POST提交到submit.php中了。使用$_POST变量可以获取这些变量的值，与$_GET变量不同的是，它们不会显示在URL中。这些变量的获取代码如下：

```
$openid        = $_POST["openid"];
$name         = $_POST["name"];
$sex          = $_POST["sex"];
$age          = $_POST["age"];
$mobile               = $_POST["mobile"];
$bookdate     = $_POST["bookdate"];
$bookexpert   = $_POST["bookexpert"];
```

我们通过PHPMailer将订单中的内容发送到邮件中，PHPMailer是PHP语言下的一个功能全面的电子邮件类，其官方网址为：http://phpmailer.worxware.com/。

使用PHPMailer发送订单的代码如下所示：

```
function sendMail()
{
    global $openid;
    global $name;
    global $sex;
    global $age;
    global $mobile;
    global $bookdate;
    global $bookexpert;

    $Subject = "微信订单";
    $receiver = "yourreceiver@qq.com";
    $content = "姓名：".$name."\n".
    "性别：".$sex."\n".
    "年龄：".$age."\n".
    "手机：".$mobile."\n".
    "预约日期：".$bookdate."\n".
    "预约专家：".$bookexpert."\n".
    "微信ID：".$openid;

    require_once('phpmailer/class.phpmailer.php');
    $mail = new PHPMailer();
    $mail->IsSMTP();
    $mail->CharSet = "utf-8";
    $mail->Host = "smtp.163.com";
    $mail->SMTPAuth = true;
    $mail->Username = "youraccount@163.com";
    $mail->Password = "yourpassword";
    $mail->From = "youraccount@163.com";
    $mail->FromName = "微信订单";
    $mail->AddAddress($receiver, "");
    $mail->Subject = $Subject;
    $mail->Body = $content;
    if(!$mail->Send()){
        return '提交失败！'.$mail->ErrorInfo;
    }else{
        return '提交成功';
    }
}
```

在上述代码中，需要配置SMTP服务器，启用SMTP认证，并且配置账号、密码、收件人邮箱、邮件标题及内容，最后使用send（）函数发送邮件。

邮件发送的结果和其他订单信息将一起在页面中显示出来，实现代码如下所示：

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset= UTF-8">
    <title>预约口腔医生</title>
    <meta name="viewport" content="width=device-width,height=device-height,
inital-scale=1.0,maximum-scale=1.0,user-scalable=no;">
    <meta name="apple-mobile-web-app-capable" content="yes">
     <meta name="apple-mobile-web-app-status-bar-style" content= "black">
    <meta name="format-detection" content="telephone=no">
    <link href="css/order.css" rel="stylesheet" type="text/css">
    <script type="text/javascript" src="js/jquery.min.js"></script>
    <script type="text/javascript" src="js/main.js"></script>
  </head>

  <body id="wrap">
    <div class="banner">
      <div id="wrapper">
        <div id="scroller">
          <ul id="thelist">
            <li>
              <img src="img/logo.png" alt="">
            </li>
          </ul>
        </div>
      </div>
      <div class="clr"></div>
    </div>
    <div class="cardexplain">
      <ul class="round roundyellow" id="success" >
        <li>
<?php echo $result;?></li>
      </ul>
      <ul class="round">
        <li class="title mb"><span class="none">您提交的信息</span></li>
        <li class="nob">
          <table width="100%" border="0" cellspacing="0" cellpadding="0" class="kuang">
            <tbody>
              <tr>
                <th>姓名</th>
                <td><?php echo $name;?></td>
              </tr>
            </tbody>
          </table>
        </li>
        <li class="nob">
          <table width="100%" border="0" cellspacing="0" cellpadding= "0" class="kuang">
            <tbody>
              <tr>
                <th>性别</th>
                <td><?php echo $sex;?></td>
              </tr>
            </tbody>
          </table>
        </li>
        <li class="nob">
          <table width="100%" border="0" cellspacing="0" cellpadding= "0" class="kuang">
```

```
            <tbody>
              <tr>
                <th>年龄</th>
                <td><?php echo $age;?></td>
              </tr>
            </tbody>
          </table>
        </li>
        <li class="nob">
          <table width="100%" border="0" cellspacing="0" cellpadding= "0" class="kuang">
            <tbody>
              <tr>
                <th>手机</th>
                <td><?php echo $mobile;?></td>
              </tr>
            </tbody>
          </table>
        </li>
        <li class="nob">
          <table width="100%" border="0" cellspacing="0" cellpadding= "0" class="kuang">
            <tbody>
              <tr>
                <th>预约日期</th>
                <td><?php echo $bookdate;?></td>
              </tr>
            </tbody>
          </table>
        </li>
        <li class="nob">
          <table width="100%" border="0" cellspacing="0" cellpadding= "0" class="kuang">
            <tbody>
              <tr>
                <th>预约专家</th>
                <td><?php echo $bookexpert;?></td>
              </tr>
            </tbody>
          </table>
        </li>
      </ul>
    </div>
  </body>
</html>
```

一个发送成功的界面如图9-5所示。

图9-5　订单提交成功

如果在微信中启用了“QQ邮箱提醒”功能，且邮箱与微信账号绑定，那么可以直接在微信中收到订单提醒消息，并且可以在微信中打开，如图9-6所示。

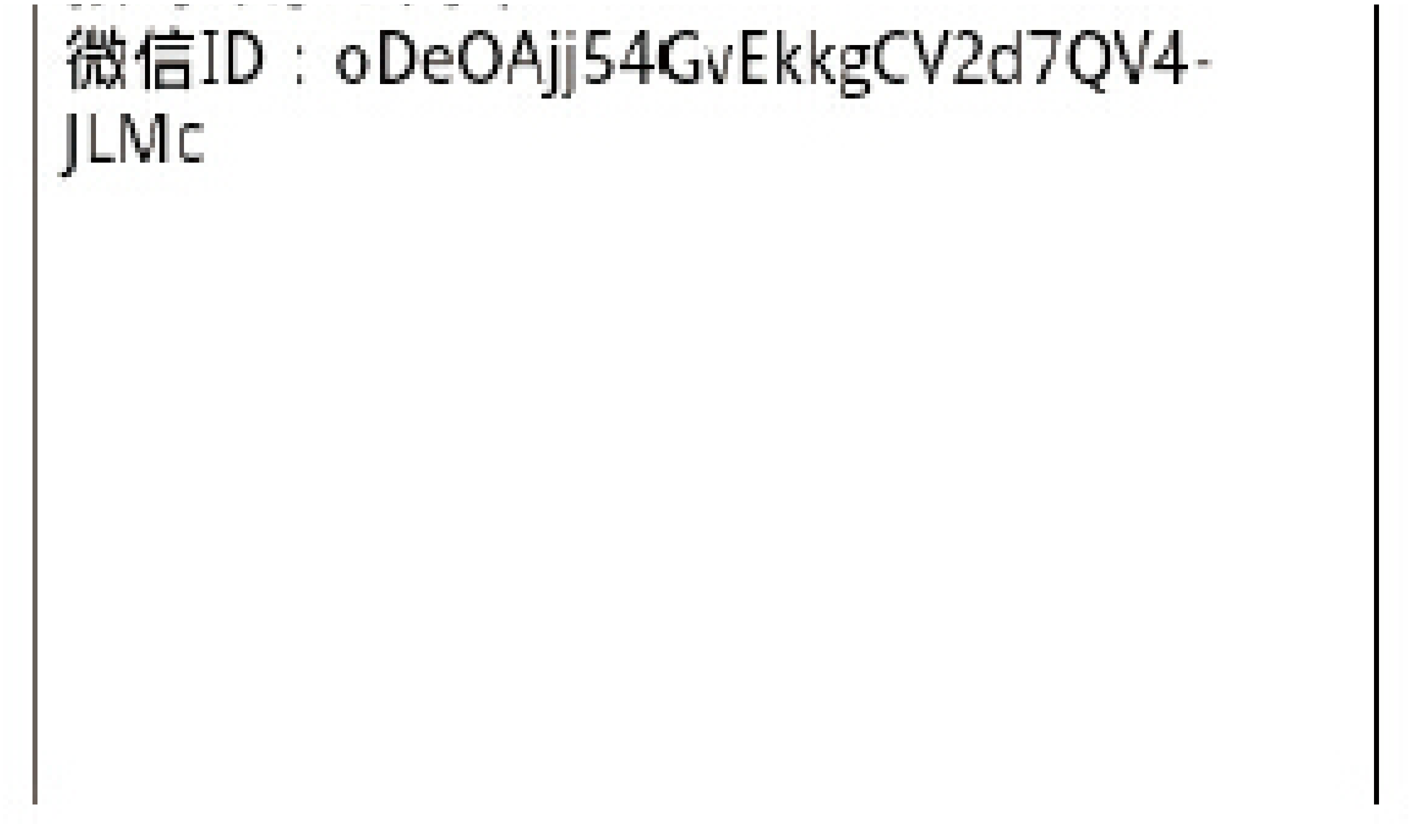


图9-6　QQ邮箱收到订单

9.3　产品相册

9.3.1　相册

对于地产、婚纱等行业来说，其产品经常是以大幅照片展示给用户，以视觉效果来吸引顾客购买，对于这些行业的微信公众账号，我们可以开发一个相册功能。

我们将使用PhotoSwipe图片库插件。PhotoSwipe是专为移动触摸设备设计的相册库，它兼容所有iPhone、iPad、Android、Blackberry以及桌面浏览器，其底层基于HTML/CSS/JavaScript实现，是一款开源的相册产品。其官方网站为http://www.photoswipe.com/。

PhotoSwipe是一个自身独立的JavaScript库，可以很方便地与网站集成，同时针对移动浏览器（webkit）进行了大量的优化。

使用PhotoSwipe开发，首先需要引用它的相关类库及样式表，而引用PhotoSwipe库之前需要先引用Klass，引用代码如下所示：

```
<link href="css/styles.css" type="text/css" rel="stylesheet"/>
<link href="css/photoswipe.css" type="text/css" rel="stylesheet"/>
<script type="text/javascript" src="js/klass.min.js"></script>
<script type="text/javascript" src="js/code.photoswipe-3.0.5.min.js"></script>
```

在JS中添加DOMContentLoaded事件监听，使用DOMContentLoaded事件的作用是只需要加载页面，而不需要加载相册中的图片等，如果图片过多、过大或者网络延迟太久，必然会导致加载时间很长，从而影响用户体验。事件监听代码如下：

```
document.addEventListener('DOMContentLoaded', function(){
}, false);
```

再在JavaScript中定义选项变量options，该变量用于配置PhotoSwipe，这里主要配置了获取图片地址和标题的方法，代码如下所示：

```
var options = {
    preventHide: true,
    getImageSource: function(obj){
        return obj.url;
       },
    getImageCaption: function(obj){
        return obj.caption;
       }
   },
```

PhotoSwipe可以配置的选项如下所示：

· allowUserZoom：允许用户双击放大/移动方式查看图片。默认值为true。

· autoStartSlideshow：当PhotoSwipe激活后，自动播放幻灯片。默认值为false。

· allowRotationOnUserZoom：只有iOS支持用户在缩放/平移模式下用手势旋转图像。默认值为false。

· backButtonHideEnabled：按返回键隐藏相册幻灯片，主要是Android和Blackberry使用，支持BB6、Androidv2.1、iOS4以及更新版本。默认值为true。

· captionAndToolbarAutoHideDelay：标题栏和工具栏自动隐藏的延迟时间。默认值为5000（毫秒）。如果设为0则不会自动隐藏（可按Tab键切换）。

· captionAndToolbarFlipPosition：标题栏和工具栏切换位置（让caption显示在底部而toolbar显示在顶部）。默认值为false。

· captionAndToolbarHide：隐藏标题栏和工具栏。默认值为false。

· captionAndToolbarOpacity：标题栏和工具栏的透明度（0~1）。默认值为0.8。

· captionAndToolbarShowEmptyCaptions：即使当前图片的标题是空，也显示标题栏。默认值为true。

· cacheMode：缓存模式，Code.PhotoSwipe.Cache.Mode.normal（正常）或者Code.PhotoSwipe.

· Cache.Mode.aggressive（激进）。决定PhotoSwipe如何管理图片缓存。

· Aggressive：模式将会积极地设置非当前、上一张、下一张的图片为空的类型，对于老版本iOS浏览器下的大图片内存溢出很有用。大多数情况下，normal模式就可以了。

· doubleTapSpeed：双击的最大间隔。默认值为300（毫秒）。

· doubleTapZoomLevel：当用户双击的时候放大的倍数，即默认的"zoom-in"（拉近）级别。默认值为2.5。

· enableDrag：允许拖动上一张/下一张图片到当前界面。默认值为true。

· enableKeyboard：允许键盘操作（左右箭头切换、Esc退出、Enter自动播放、空格键显示/隐藏标题栏、退出）。默认值为true。

· enableMouseWheel：允许鼠标滚轮操作。默认值为true。

· fadeInSpeed：淡入效果元素的速度（持续时间），单位为毫秒。默认值为250。

· fadeOutSpeed：淡出效果元素的速度（持续时间），单位为毫秒。默认值为250。

· imageScaleMethod：图片缩放方法（模式）。可选值："fit"、"fitNoUpscale"和"zoom"。

· 模式"fit"保证图像适应屏幕。"fitNoUpscale"和"fit"类似但是不会放大图片。"zoom"将图片全屏，但有可能图片缩放不是等比例的。默认值为"fit"。

· invertMouseWheel：反转鼠标滚轮。默认情况下，鼠标向下滚动将切换到下一张，向上则切换到上一张。默认值为false。

· jQueryMobile：指示PhotoSwipe是否集成进了jQueryMobile项目。默认情况下，PhotoSwipe将会自动判断是否集成。

· jQueryMobileDialogHash：jQueryMobile的window、dialog页面所使用的hash标签。默认值为"&ui-state=dialog"。

· loop：相册是否自动循环。默认值为true。

· margin：两张图之间的间隔，单位是像素。默认值为20。

· maxUserZoom：最大放大倍数。默认值为5.0（设置为0将被忽略）。

· minUserZoom：图像最小的缩小倍数。默认值为0.5（设置为0将会忽略）。

· mouseWheelSpeed：响应鼠标滚轮的灵敏度。默认值为500（毫秒）。

· nextPreviousSlideSpeed：当单击上一张、下一张按钮后，延迟多少毫秒执行切换。默认值为0（立即切换）。

· preventHide：阻止用户关闭PhotoSwipe，同时也会隐藏工具栏上的关闭按钮。在独享的页面使用。默认值为false。

· preventSlideshow：阻止自动播放模式，同时也会隐藏工具栏里的播放按钮。默认值为false。

· slideshowDelay：自动播放模式下多长时间播放下一张。默认值为3000（毫秒）。

· slideSpeed：图片滑进视图的时间。默认值为250（毫秒）。

· swipeThreshold：手指滑动多少像素才触发一个swipe手势事件。默认值为50。

· swipeTimeThreshold：定义触发swipe（滑动）手势的最大毫秒数，太慢了则不会触发滑动，只会拖动当前照片的位置。默认值为250。

· slideTimingFunction：滑动时的Easingfunction。默认值为"ease-out"。

· zIndex：初始的zIndex值。默认值为1000。

· enableUIWebViewRepositionTimeout：检查设备的方向是否改变。默认值为false。

· uiWebViewResetPositionDelay：定时检查设备的方向是否改变的时间。默认值为500（毫秒）。

· preventDefaultTouchEvents：阻止默认的touch事件，比如页面滚动。默认值为true。

· target：必须是一个合法的DOM元素（如DIV）。默认是window（全页面）。而如果是某个低级别的DOM，则在DOM内显示，可能非全屏。

然后实例化一个PhotoSwipe对象，并为该对象指定相册中各图片的链接和标题，相关代码如下所示：

```
instance = PhotoSwipe.attach(
   [
       { url: 'imghx/image0132.jpg', caption: 'A户型二室二厅一卫 80.07平方米'},
       { url: 'imghx/image0092.jpg', caption: 'B户型二室二厅一卫 84.65平方米'},
       { url: 'imghx/image0032.jpg', caption: 'C户型二室二厅二卫 103.89平方米'},
   ],
options
);
```

最后，使用show（）方法设置相册首页图片的索引，代码如下所示：

```
instance.show(0);
```

完整的代码如下所示：

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
    <title>户型鉴赏</title>
    <META charset=utf-8>
    <meta name="author" content="Ste Brennan - Code Computerlove - http://www.codecomputerlove.com/" />
    <meta content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0;" name="viewport" />
    <meta name="apple-mobile-web-app-capable" content="yes" />

    <link href="css/styles.css" type="text/css" rel="stylesheet" />
    <link href="css/photoswipe.css" type="text/css" rel="stylesheet" />
    <script type="text/javascript" src="js/klass.min.js"></script>
    <script type="text/javascript" src="js/code.photoswipe-3.0.5.min.js"></script>

    <script type="text/javascript">
        (function(window, PhotoSwipe){
            document.addEventListener('DOMContentLoaded', function(){

                var options = {
                        preventHide: true,
                        getImageSource: function(obj){
                            return obj.url;
                        },
                        getImageCaption: function(obj){
                            return obj.caption;
                        }
                    },
                    instance = PhotoSwipe.attach(
                        [
                            { url: 'imghx/image013.jpg', caption: 'A户型 二室二厅一卫 80.07平方米'},
                            { url: 'imghx/image009.jpg', caption: 'B户型 二室二厅一卫 84.65平方米'},
                            { url: 'imghx/image003.jpg', caption: 'C户型 二室二厅二卫 103.89平方米'},
                        ],
                        options
                    );

                    instance.show(0);

            }, false);
        }(window, window.Code.PhotoSwipe));

    </script>
</head>
<body>
<script type="text/javascript">
     document.addEventListener('WeixinJSBridgeReady', function onBridgeReady() {
        WeixinJSBridge.call('hideOptionMenu');
    });
     document.addEventListener('WeixinJSBridgeReady', function onBridgeReady() {
        WeixinJSBridge.call('hideToolbar');
    });
</script>
</body>
</html>
```

上述代码执行后，效果如图9-7所示。

图9-7　户型图片展示

9.3 产品相册

9.3.1 相册

对于地产、婚纱等行业来说，其产品经常是以大幅照片展示给用户，以视觉效果来吸引顾客购买，对于这些行业的微信公众账号，我们可以开发一个相册功能。

我们将使用PhotoSwipe图片库插件。PhotoSwipe是专为移动触摸设备设计的相册库，它兼容所有iPhone、iPad、Android、Blackberry以及桌面浏览器，其底层基于HTML/CSS/JavaScript实现，是一款开源的相册产品。其官方网站为http://www.photoswipe.com/。

PhotoSwipe是一个自身独立的JavaScript库，可以很方便地与网站集成，同时针对移动浏览器（webkit）进行了大量的优化。

使用PhotoSwipe开发，首先需要引用它的相关类库及样式表，而引用PhotoSwipe库之前需要先引用Klass，引用代码如下所示：

```
<link href="css/styles.css" type="text/css" rel="stylesheet"/>
<link href="css/photoswipe.css" type="text/css" rel="stylesheet"/>
<script type="text/javascript" src="js/klass.min.js"></script>
<script type="text/javascript" src="js/code.photoswipe-3.0.5.min.js"></script>
```

在JS中添加DOMContentLoaded事件监听，使用DOMContentLoaded事件的作用是只需要加载页面，而不需要加载相册中的图片等，如果图片过多、过大或者网络延迟太久，必然会导致加载时间很长，从而影响用户体验。事件监听代码如下：

```
document.addEventListener('DOMContentLoaded', function(){
}, false);
```

再在JavaScript中定义选项变量options，该变量用于配置PhotoSwipe，这里主要配置了获取图片地址和标题的方法，代码如下所示：

```
var options = {
    preventHide: true,
    getImageSource: function(obj){
        return obj.url;
      },
    getImageCaption: function(obj){
        return obj.caption;
      }
   },
```

PhotoSwipe可以配置的选项如下所示：

· allowUserZoom：允许用户双击放大/移动方式查看图片。默认值为true。

· autoStartSlideshow：当PhotoSwipe激活后，自动播放幻灯片。默认值为false。

· allowRotationOnUserZoom：只有iOS支持用户在缩放/平移模式下用手势旋转图像。默认值为false。

· backButtonHideEnabled：按返回键隐藏相册幻灯片，主要是Android和Blackberry使用，支持BB6、Androidv2.1、iOS4以及更新版本。默认值为true。

· captionAndToolbarAutoHideDelay：标题栏和工具栏自动隐藏的延迟时间。默认值为5000（毫秒）。如果设为0则不会自动隐藏（可按Tab键切换）。

· captionAndToolbarFlipPosition：标题栏和工具栏切换位置（让caption显示在底部而toolbar显示在顶部）。默认值为false。

· captionAndToolbarHide：隐藏标题栏和工具栏。默认值为false。

· captionAndToolbarOpacity：标题栏和工具栏的透明度（0~1）。默认值为0.8。

· captionAndToolbarShowEmptyCaptions：即使当前图片的标题是空，也显示标题栏。默认值为true。

· cacheMode：缓存模式，Code.PhotoSwipe.Cache.Mode.normal（正常）或者Code.PhotoSwipe.

· Cache.Mode.aggressive（激进）。决定PhotoSwipe如何管理图片缓存。

· Aggressive：模式将会积极地设置非当前、上一张、下一张的图片为空的类型，对于老版本iOS浏览器下的大图片内存溢出很有用。大多数情况下，normal模式就可以了。

· doubleTapSpeed：双击的最大间隔。默认值为300（毫秒）。

· doubleTapZoomLevel：当用户双击的时候放大的倍数，即默认的“zoom-in”（拉近）级别。默认值为2.5。

· enableDrag：允许拖动上一张/下一张图片到当前界面。默认值为true。

· enableKeyboard：允许键盘操作（左右箭头切换、Esc退出、Enter自动播放、空格键显示/隐藏标题栏、退出）。默认值为true。

· enableMouseWheel：允许鼠标滚轮操作。默认值为true。

· fadeInSpeed：淡入效果元素的速度（持续时间），单位为毫秒。默认值为250。

· fadeOutSpeed：淡出效果元素的速度（持续时间），单位为毫秒。默认值为250。

· imageScaleMethod：图片缩放方法（模式）。可选值：“fit”、“fitNoUpscale”和“zoom”。

· 模式“fit”保证图像适应屏幕。“fitNoUpscale”和“fit”类似但是不会放大图片。“zoom”将图片全屏，但有可能图片缩放不是等比例的。默认值为“fit”。

· invertMouseWheel：反转鼠标滚轮。默认情况下，鼠标向下滚动将切换到下一张，向上则切换到上一张。默认值为false。

· jQueryMobile：指示PhotoSwipe是否集成进了jQueryMobile项目。默认情况下，PhotoSwipe将会自动判断是否集成。

· jQueryMobileDialogHash：jQueryMobile的window、dialog页面所使用的hash标签。默认值为“&ui-state=dialog”。

· loop：相册是否自动循环。默认值为true。

· margin：两张图之间的间隔，单位是像素。默认值为20。

· maxUserZoom：最大放大倍数。默认值为5.0（设置为0将被忽略）。

· minUserZoom：图像最小的缩小倍数。默认值为0.5（设置为0将会忽略）。

· mouseWheelSpeed：响应鼠标滚轮的灵敏度。默认值为500（毫秒）。

· nextPreviousSlideSpeed：当单击上一张、下一张按钮后，延迟多少毫秒执行切换。默认值为0（立即切换）。

· preventHide：阻止用户关闭PhotoSwipe，同时也会隐藏工具栏上的关闭按钮。在独享的页面使用。默认值为false。

· preventSlideshow：阻止自动播放模式，同时也会隐藏工具栏里的播放按钮。默认值为false。

· slideshowDelay：自动播放模式下多长时间播放下一张。默认值为3000（毫秒）。

· slideSpeed：图片滑进视图的时间。默认值为250（毫秒）。

· swipeThreshold：手指滑动多少像素才触发一个swipe手势事件。默认值为50。

· swipeTimeThreshold：定义触发swipe（滑动）手势的最大毫秒数，太慢了则不会触发滑动，只会拖动当前照片的位置。默认值为250。

· slideTimingFunction：滑动时的Easingfunction。默认值为“ease-out”。

· zIndex：初始的zIndex值。默认值为1000。

· enableUIWebViewRepositionTimeout：检查设备的方向是否改变。默认值为false。

· uiWebViewResetPositionDelay：定时检查设备的方向是否改变的时间。默认值为500（毫秒）。

· preventDefaultTouchEvents：阻止默认的touch事件，比如页面滚动。默认值为true。

· target：必须是一个合法的DOM元素（如DIV）。默认是window（全页面）。而如果是某个低级别的DOM，则在DOM内显示，可能非全屏。

然后实例化一个PhotoSwipe对象，并为该对象指定相册中各图片的链接和标题，相关代码如下所示：

```
instance = PhotoSwipe.attach(
   [
       { url: 'imghx/image0132.jpg', caption: 'A户型二室二厅一卫 80.07平方米'},
       { url: 'imghx/image0092.jpg', caption: 'B户型二室二厅一卫 84.65平方米'},
       { url: 'imghx/image0032.jpg', caption: 'C户型二室二厅二卫 103.89平方米'},
   ],
options
);
```

最后，使用show（）方法设置相册首页图片的索引，代码如下所示：

```
instance.show(0);
```

完整的代码如下所示：

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
    <title>户型鉴赏</title>
    <META charset=utf-8>
    <meta name="author" content="Ste Brennan - Code Computerlove - http://www.codecomputerlove.com/" />
    <meta content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0;" name="viewport" />
    <meta name="apple-mobile-web-app-capable" content="yes" />

    <link href="css/styles.css" type="text/css" rel="stylesheet" />
    <link href="css/photoswipe.css" type="text/css" rel="stylesheet" />
    <script type="text/javascript" src="js/klass.min.js"></script>
    <script type="text/javascript" src="js/code.photoswipe-3.0.5.min.js"></script>

    <script type="text/javascript">
        (function(window, PhotoSwipe){
            document.addEventListener('DOMContentLoaded', function(){

                var options = {
                        preventHide: true,
                        getImageSource: function(obj){
                            return obj.url;
                        },
                        getImageCaption: function(obj){
                            return obj.caption;
                        }
                    },
                    instance = PhotoSwipe.attach(
                        [
                            { url: 'imghx/image013.jpg', caption: 'A户型 二室二厅一卫 80.07平方米'},
                            { url: 'imghx/image009.jpg', caption: 'B户型 二室二厅一卫 84.65平方米'},
                            { url: 'imghx/image003.jpg', caption: 'C户型 二室二厅二卫 103.89平方米'},
                        ],
                        options
                    );

                    instance.show(0);

            }, false);
        }(window, window.Code.PhotoSwipe));

    </script>
</head>
<body>
<script type="text/javascript">
     document.addEventListener('WeixinJSBridgeReady', function onBridgeReady() {
        WeixinJSBridge.call('hideOptionMenu');
    });
     document.addEventListener('WeixinJSBridgeReady', function onBridgeReady() {
        WeixinJSBridge.call('hideToolbar');
    });
</script>
</body>
```

```
</html>
```

上述代码执行后，效果如图9-7所示。

图9-7　户型图片展示

9.3.2　画廊

除了直接显示图片之外，还可以使用画廊的功能，先显示所有图片列表，再以幻灯片的方式显示单个图片，使用jQuery Mobile显示画廊的代码如下所示。

```
<!DOCTYPE html>
<html>
  <head>
    <title>PhotoSwipe</title>
     <link rel="stylesheet" href="http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../lib/jquery.mobile-1.0a4.1.min.cs
     <link href="http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../photoswipe.css" type="text/css" rel="stylesheet"
     <style type="text/css">
             div.gallery-row:after { clear: both; content: "."; display: block; height: 0; visibility: hidden; }
             div.gallery-item { float: left; width: 33.333333%; }
             div.gallery-item a { display: block; margin: 5px; border: 1px solid #3c3c3c; }
             div.gallery-item img { display: block; width: 100%; height: auto; }
             #Gallery1 .ui-content, #Gallery2 .ui-content { overflow: hidden; }
     </style>
     <script type="text/javascript" src="http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../lib/jquery-1.6.1.min.js">
     <script type="text/javascript" src="http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../lib/jquery.mobile-1.0a4.1
     <script type="text/javascript" src="http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../lib/simple-inheritance.mi
     <script type="text/javascript" src="http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../lib/jquery.animate-enhanc
     <script type="text/javascript" src="http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/15214/OEBPS/Text/../code-photoswipe-jQuery-1.
    <script type="text/javascript">
      /*
      This example shows how to set up PhotoSwipe using jQuery Mobile.
      */
      $(document).ready(function(){
        $("#Gallery1 a").photoSwipe();
      });
    </script>
  </head>
  <body>
    <div data-role="page" id="Gallery1" class="gallery-page">
      <div data-role="header">
        <h1>First Gallery</h1>
      </div>
      <div data-role="content">
        <div class="gallery">
          <div class="gallery-row">
            <div class="gallery-item"><a href="images/full/001.jpg" rel="external"><img src="images/thumb/001.jpg" alt="Image 001" /></a></div>
            <div class="gallery-item"><a href="images/full/002.jpg" rel= "external"><img src="images/thumb/002.jpg" alt="Image 002" /></a></div>
            <div class="gallery-item"><a href="images/full/003.jpg" rel= "external"><img src="images/thumb/003.jpg" alt="Image 003" /></a></div>
          </div>
          <div class="gallery-row">
            <div class="gallery-item"><a href="images/full/004.jpg" rel= "external"><img src="images/thumb/004.jpg" alt="Image 004" /></a></div>
            <div class="gallery-item"><a href="images/full/005.jpg" rel= "external"><img src="images/thumb/005.jpg" alt="Image 005" /></a></div>
            <div class="gallery-item"><a href="images/full/006.jpg" rel= "external"><img src="images/thumb/006.jpg" alt="Image 006" /></a></div>
          </div>
          <div class="gallery-row">
```

```
            <div class="gallery-item"><a href="images/full/007.jpg" rel= "external"><img src="images/thumb/007.jpg" alt="Image 007" /></a></div>
            <div class="gallery-item"><a href="images/full/008.jpg" rel= "external"><img src="images/thumb/008.jpg" alt="Image 008" /></a></div>
            <div class="gallery-item"><a href="images/full/009.jpg" rel= "external"><img src="images/thumb/009.jpg" alt="Image 009" /></a></div>
          </div>
          <div class="gallery-row">
            <div class="gallery-item"><a href="images/full/010.jpg" rel= "external"><img src="images/thumb/010.jpg" alt="Image 010" /></a></div>
            <div class="gallery-item"><a href="images/full/011.jpg" rel= "external"><img src="images/thumb/011.jpg" alt="Image 011" /></a></div>
            <div class="gallery-item"><a href="images/full/012.jpg" rel= "external"><img src="images/thumb/012.jpg" alt="Image 012" /></a></div>
          </div>
        </div>
      </div>
      <div data-role="footer">
        <h4>&copy; 2011 Code Computerlove</h4>
      </div>
    </div>
  </body>
</html>
```

上述代码使用jQuery Mobile创建了页面的三个部分：header、content、footer。其中content中放置了4行3列的图片列表，使用内部样式表控制显示。使用jQuery的read（）方法来启动PhotoSwipe。执行代码后效果如图9-8所示。

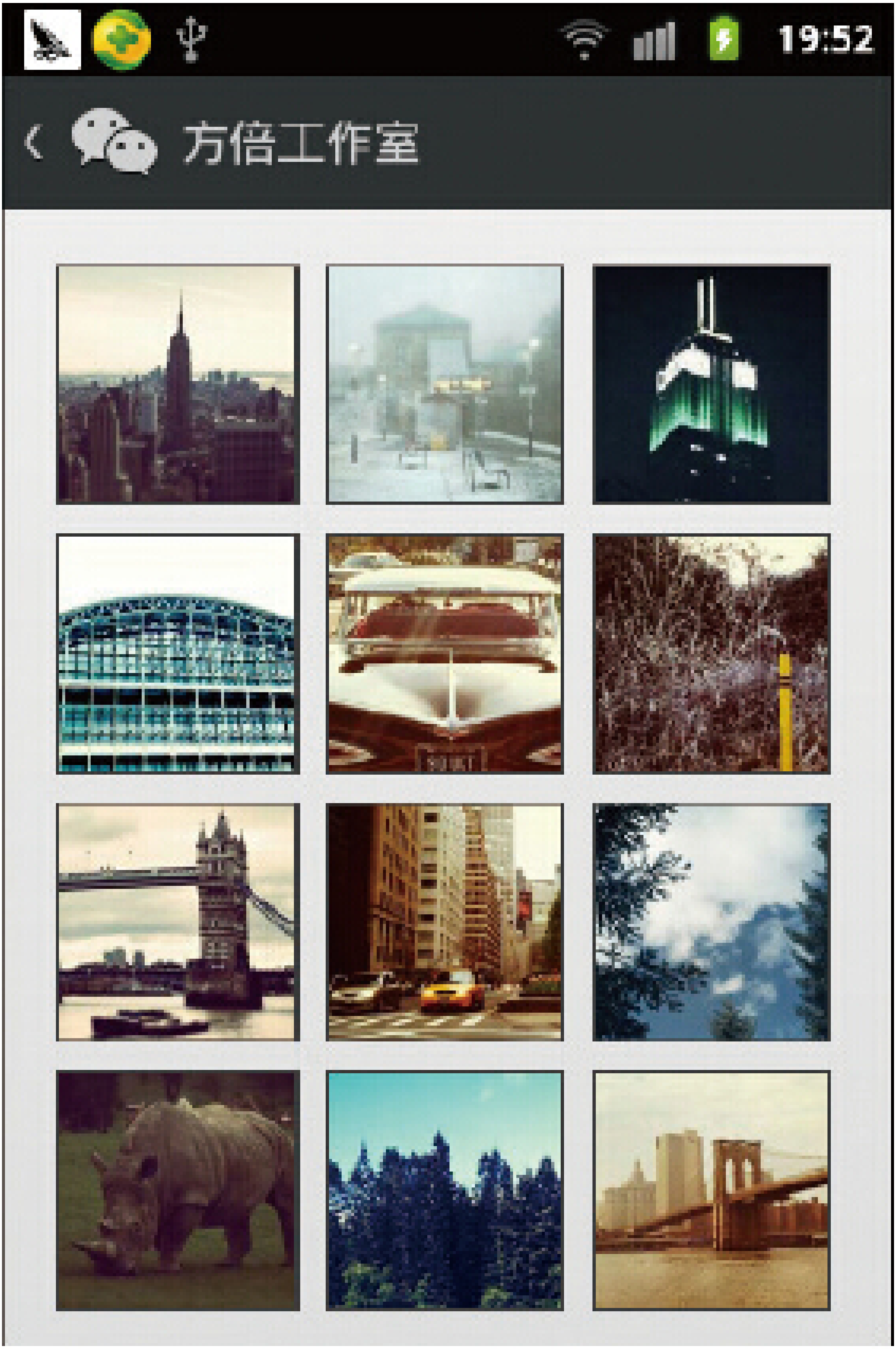

19:52
方倍工作室

图9-8 jQuery Mobile集成PhotoSwipe

9.4 大转盘

为了加强与用户的互动，很多微信公众账号配置了抽奖等营销活动功能，大转盘就是最常用的一种。相比传统的广告推广方式，大转盘互动性强、趣味性高，深受用户的喜爱。本节介绍微信公众平台上大转盘的实现原理。

9.4.1 页面布局

大转盘开发最重要的素材是转盘图片和指针图片，如图9-9所示。

图9-9 转盘和指针

接着需要实现页面布局，其中将转盘设置为前景，指针设置为背景，以此定义外部、内部两个容器，HTML代码如下：

```
<div id="outercont"  >
  <div id="outer-cont">
    <div id="outer"><img src="img/activity-lottery-5.png"></div>
  </div>
  <div id="inner-cont">
    <div id="inner"><img src="img/activity-lottery-2.png"></div>
  </div>
</div>
```

相应的CSS控制代码如下：

```
#outer-cont {
   position: relative;
   width: 100%;
   top: 20px;
   margin-bottom: 30px;
}
#inner-cont {
  position: absolute;
```

```
  width: 100%;
  top: 70px;
}
#outer {
  height: 227px;
  margin: 0 auto;
  max-width: 227px;
  width: 227px;
}
#inner {
  cursor: pointer;
  height: 110px;
  margin: 0 auto;
  max-width: 90px;
  width: 90px;
}
#outer img, #inner img {
  display: block;
  margin: 0 auto;
}
```

再定义一些要显示的内容区域，主要有中奖结果区、奖项设置区、活动说明区。相关代码如下所示：

```
<div class="content"  >
  <div class="boxcontent boxyellow" id="result" >
    <div class="box">
      <div class="title-orange"><span>恭喜中奖</span></div>
      <div class="Detail">
        <p>您中了 <span class="red" id="prizelevel" ></span></p>
        <p>奖品为 <span class="red" id="prizename"></span></p>
      </div>
    </div>
  </div>
  <div class="boxcontent boxyellow">
    <div class="box">
      <div class="title-green"><span>奖项设置：</span></div>
      <div class="Detail">
        <p>一等奖：iPhone 6 </p>
        <p>二等奖：iPhone 5S</p>
        <p>三等奖：iPhone 5C</p>
        <p>四等奖：iPad Air </p>
        <p>五等奖：iPhone 4S</p>
        <p>六等奖：iPad mini</p>
      </div>
    </div>
  </div>
  <div class="boxcontent boxyellow">
    <div class="box">
      <div class="title-green">活动说明：</div>
      <div class="Detail">
        <p>本次活动每人可以抽奖5次 </p>
        <p>祝您中奖！</p>
      </div>
    </div>
  </div>
</div>
```

在初始页面时，中奖结果区域通过样式表控制成不可见，抽奖页面的最终实现效果如图9-10所示。

图9-10　大转盘

9.4.2　开发实现

下面，我们需要实现抽奖算法及中奖通知的功能。

首先定义指针转动时中奖的角度和未中奖的角度，以便判断用户是否中奖，图中有6个奖项，相应地也有6个未中奖区域，奖项角度及其他变量定义如下：

```
vartotalAngle  = 0;
var steps = [];
var loseAngle = [36, 96, 156, 216, 276, 336];
var winAngle = [6, 66, 126, 186, 246, 306];
var prizeLevel;
var now = 0;
var count = 0;
var a = 0.01;
var outter, inner, timer, running = false;
```

用户进入抽奖界面后，将单击抽奖指针，相应的代码如下：

```
$("#inner").click(function() {
  if (running) return;
  if (count >= 3) {
    alert("达到最大抽奖次数！");
    return
  }
  $.ajax({
    url: "data.php",
    dataType: "json",
    data: {
      openid: "<?php echo $_GET["openid"];?>",
      time: (new Date()).valueOf()
    },
    beforeSend: function() {
      running = true;
      timer = setInterval(function() {
        i += 5
      },
      1)
    },
    success: function(data) {
      //达到最大抽奖次数
      if (data.status == "MAX") {
        alert("您已达到最大抽奖次数！");
        count = 3;
        clearInterval(timer);
        return
      }
      //有中奖时转盘转到相应位置
      if (data.status == "WIN") {
        $("#prizename").text(data.prizename);
        count = 3;
        clearInterval(timer);
        prizeLevel = data.prizelevel;
        start(winAngle[data.prizelevel - 1]);
        return
      }
      //未中奖则再给机会
      running = false;
```

```
40       count++
41       prizeLevel = null;
42       start()
43     },
44     //未获取JSON返回，前台处理
45     error: function() {
46       prizeLevel = null;
47       start();
48       running = false;
49       count++
50     },
51     timeout: 4000
52   })
53 })
```

单击事件发生时，页面将向data.php文件发送POST请求，将当前用户的openid和时间传递过去。data.php在收到数据后将需要进行一系列复杂的处理，至少包括：

· 判断该用户是否还有抽奖权限。

· 写入当前用户抽奖记录。

· 判断奖项是否仍有剩余。

· 随机生成本次中奖概率。

· 分配本次中奖奖项。

· 返回本次中奖结果。

下面是一个中奖情形的返回结果：

```
echo '{"status": "WIN", "prizename": "iPhone 5S", "prizelevel": "2"}';
```

该情形表示当前已中奖，奖品等级是2，奖品为iPhone 5S。data.php将该JSON数据返回给请求页面，原页面收到数据后，第29行~第37行代码将进行处理，它将抽奖次数直接置为最大抽奖次数3，并且计算转盘将要旋转的角度，而这个旋转角度也就是奖项的角度，以此确保转盘停止后，指针落点无误。

而当中奖数据返回中没有要中奖的标记（第38行~第42行）或者没有接收到返回的JSON数据（第44行~第50行）时，转盘也需要计算旋转角度，页面也将在前台累加抽奖次数，直到达到最大抽奖次数，然后提示用户抽奖次数已经用完了。如果返回的JSON数据中也显示已经达到最大抽奖次数，则以JSON数据标识优先判断（第22行~第28行）。

当启动转盘转动时，本次抽奖结果已经被旋转角度确定下来了，这是通过start（）方法来实现的，该方法带有一个参数deg。如果中奖，则deg传输进来的时候就已经是某奖项的角度；如果没有中奖，deg直接传空，这时将随机计算出一个非奖项的角度并赋给它。start（）方法的实现如下所示：

```
function start(deg) {
    deg = deg || loseAngle[parseInt(loseAngle.length * Math.random())];
    running = true;
    clearInterval(timer);
    totalAngle  = 360 * 5 + deg;
    steps = [];
    now = 0;
    countSteps();
    requestAnimFrame(step)
}
```

在start（）函数中，需要根据旋转角度生成本次转动的步骤，这时往往需要另外添加N个360度，以便在旋转N圈后再落到真正的奖项区域。而这个旋转角度则以数组的方式保存。其角度差需要逐渐变小，以实现减速旋转最终停下来的效果。该数组的生成代码如下所示：

```
function countSteps() {
    var t = Math.sqrt(2 * totalAngle / a);
    var v = a * t;
    for (var i = 0; i < t; i++) {
        steps.push((2 * v * i - a * i * i) / 2)
    }
    steps.push(totalAngle)
}
```

上面代码将生成一个元素个数非常大的角度列表数组，数组存储到steps中。本书为了简便，生成了一个简化版本，如表9-1所示。

表9-1　旋转角度数组

停留角度	角度差	停留角度	角度差
0.0000000000000	4.5156501172304	50.9347517584563	2.1156501172304
4.5156501172304	4.3556501172304	53.0504018756867	1.9556501172304
8.8713002344608	4.1956501172304	55.0060519929171	1.7956501172305
13.0669503516912	4.0356501172304	56.8017021101476	1.6356501172304
17.1026004689216	3.8756501172305	58.4373522273780	1.4756501172304
20.9782505861521	3.7156501172304	59.9130023446084	1.3156501172304
24.6939007033825	3.5556501172304	61.2286524618388	1.1556501172305
28.2495508206129	3.3956501172304	62.3843025790693	0.9956501172304
31.6452009378433	3.2356501172305	63.3799526962997	0.8356501172304
34.8808510550738	3.0756501172304	64.2156028135301	0.6756501172304
37.9565011723042	2.9156501172304	64.8912529307605	0.5156501172305
40.8721512895346	2.7556501172304	65.4069030479910	0.3556501172304
43.6278014067650	2.5956501172304	65.7625531652214	0.1956501172304
46.2234515239954	2.4356501172305	65.9582032824518	0.0000000000000
48.6591016412259	2.2756501172304		

从表9-1中可以看到，转盘将从0度旋转到65.9度，每一步的旋转角度差将越来越小，直到为0，这时将停留到预先生成的角度上。

而旋转的动画效果则是使用HTML5中的window.requestAnimFrame方法实现的，代码如下所示：

```
window.requestAnimFrame = (function() {
  return window.requestAnimationFrame ||
         window.webkitRequestAnimationFrame ||
         window.mozRequestAnimationFrame ||
         window.oRequestAnimationFrame ||
         window.msRequestAnimationFrame ||
  function(callback) {
    window.setTimeout(callback, 1000 / 60)
  }
})();
```

当动画将每个旋转角度走完以后，需要将中奖结果提示给用户。没有中奖时只需要生成一个弹出消息框，而中奖时则需要隐藏转盘、显示中奖区域、显示中奖结果。该部分代码如下所示：

```
function step() {
    outter.style.webkitTransform = 'rotate(' + steps[now++] + 'deg)';
    outter.style.MozTransform = 'rotate(' + steps[now++] + 'deg)';
    outter.style.oTransform = 'rotate(' + steps[now++] + 'deg)';
    outter.style.msTransform = 'rotate(' + steps[now++] + 'deg)';
    if (now < steps.length) {
        requestAnimFrame(step)
    } else {
        running = false;
        setTimeout(function() {
            if (prizeLevel != null) {
                var levelName= new Array("", "一等奖", "二等奖", "三等奖", "四等奖", "五等奖", "六等奖")
                $("#prizelevel").text(levelName[prizeLevel]);
                $("#result").slideToggle(500);  //显示中奖区域
                $("#outercont").slideUp(500)    //隐藏转盘
            } else {
                alert("亲，继续努力哦！")
            }
        },
        200)
    }
}
```

最终，中奖结果的页面如图9-11所示。

0:53
方倍工作室
恭喜中奖
您中了 二等奖
奖品为 iPhone 5S
奖项设置：
一等奖：iPhone 6
二等奖：iPhone 5S
三等奖：iPhone 5C
四等奖：iPad Air
五等奖：iPhone 4S
六等奖：iPad mini
活动说明：
本次活动每人可以抽奖 3 次
祝您中奖！

图9-11　大转盘中奖页面

9.5　刮刮卡

同大转盘一样，刮刮卡也是备受用户欢迎的一个抽奖活动。本节介绍微信公众平台下如何实现刮刮卡。

9.5.1　页面布局

我们将使用wScratchPad来实现刮刮卡的功能。wScratchPad是jQuery上的一个插件，它能模拟出刮的行为，使你能“擦去”一层覆盖物，这层覆盖物既可以是一个图片，也可以是某种颜色涂层。

wScratchPad的用法如下述代码所示：

```
$('#elem').wScratchPad({
  width            : 210,          // 设置宽度,最好与图片宽度匹配
  height           : 100,          // 设置高度,最好与图片高度匹配
  image            : 'slide.jpg',  // 设置背景图片路径
  image2           : null,         // 设置覆盖图路径,仅当涂层为图片时可用
  color            : '#336699',    // 设置涂层颜色值,仅当涂层为颜色时可用
  overlay          : 'none',       // 设置涂层效果,可设置'none'和'lighter',
                                      当涂层为颜色时可用
  size                     : 10,          // 设置刮的宽度
  realtimePercent  : false,           刮完时更新抓痕百分比,此项可优化移动设备表现
  scratchDown              : null,        // scratchDown回调
  scratchUp        : null,         // scratchUp回调
  scratchMove              : null,        // scratcMove回调
  cursor           : null          // 设置自定义光标的路径
});
```

首先需要实现页面布局，这里定义一个“封面”，封面使用背景图填充，然后定义一个刮涂区和文字区，代码如下所示：

```
<div class="cover">
  <img src="img/scratch-card-banner.png">
  <div id="prize"> </div>
  <div id="scratchpad"></div>
</div>
```

背景图和封面需要同宽同高，而文字区和刮涂区则使用同样的样式表，以便重叠放置，CSS代码如下所示：

```
.cover{
  width: 320px;
  max-width: 480px;
  margin: 0 auto;
  position: relative;
}
.cover img{
  width: 320px;
}
#scratchpad,#prize{
  position: absolute;
  width: 150px;
  height: 40px;
  top: 140px;
  left: 105px;
  text-align: center;
  font-size: 20px;
  line-height: 40px;
  color: #333;
}
```

再定义三个内容区，分别是结果区、活动说明区、参与方式区，结果区默认隐藏。代码如下所示：

```
<div class="content">
  <div id="winprize" class="boxcontent boxwhite">
    <div class="box">
      <div class="title-red"><span>恭喜你中奖了</span></div>
      <div class="Detail">
        <p>您中了 <span class="red" id="prizelevel"></span></p>
        <p>奖品为 <span class="red" id="prizename"></span></p>
      </div>
    </div>
  </div>
  <div class="boxcontent boxwhite">
    <div class="box">
      <div class="title-brown">活动说明：
      </div>
      <div class="Detail">
        <p>活动时间：即日起至2016年12月31日</p>
        <p>奖品设置：<font color="red"><strong>iPhone 6</strong></font> 手机 1000 部</p>
      </div>
    </div>
  </div>
  <div class="boxcontent boxwhite">
    <div class="box">
      <div class="title-brown">参与方式：
      </div>
      <div class="Detail">
        <p> 第一步：关注微信公众账号 方倍工作室</p>
        <p> 第二步：回复<strong> 刮刮乐 </strong>三个汉字</p>
        <p> 第三步：点击图文消息进入刮奖页面</p>
        <p> 第四步：刮掉涂层查看中奖结果</p>
      </div>
    </div>
  </div>
</div>
```

在这个演示中，我们只设置了一个等级的奖品。页面布局完成后，效果如图9-12所示。

20:46
方倍工作室
刮刮卡
大奖刮出来
刮奖区
活动说明：
活动时间：即日起至2016年12月31日
奖品设置：iPhone 6 手机 1000 部
参与方式：
第一步：关注微信公众账号 方倍工作室

图9-12　刮刮卡页面

9.5.2　开发实现

接下来给wScratchPad对象配置属性和方法，相应代码如下所示：

```
 1 <script type="text/javascript">
 2   var display = false;
 3   var num = 0;
 4   var win = false;
 5   $(function(){
 6     $("#scratchpad").wScratchPad({
 7       width : 150,
 8       height : 40,
 9       color : "#a9a9a7",
10       scratchMove : function(e, percent){
11         num++;
12         //80%时自动清除
13         if(percent > 80){
14           this.clear();
15         }
16         //开始时请求中奖结果
17         if (num == 1) {
18           if (Math.floor(Math.random() * (100 + 1)) <  10){
19             //需要通过ajax请求判断openid以前是否中过奖
20             $getPrizeBefor = false;
21             if (!$getPrizeBefor){
22               win = true;
23               document.getElementById('prize').innerHTML = "一等奖";
24             }
25           }
26         }
27
28         //移动至少10次后，显示结果
29         if (num > 10){
30           //只显示一次
31           if (!display){
32             //根据概率显示
33             if (win){
34               $("#prizelevel").text("一等奖");
35               $("#prizename").text('iPhone 6');
36               $("#winprize").slideToggle(500);
37             }else {
38               document.getElementById('prize').innerHTML="谢谢参与";
39             }
40           }
41           display = true;
42         }
43       }
44     });
45   });
46 </script>
```

在上述代码中，将涂层的大小设置成和文字区一样，并将涂层颜色设置成银灰色（第9行），然后设置scratchMove的回调方法，该方法对应用户在刮奖时的响应动作。

第16行~第26行控制用户中奖的情况，当用户在刮第一下时，即开始计算本次中奖概率，此处设置中奖概率为10%（第18行），如果用户中奖，则需向后台服务器请求校验该用户是否以前抽过奖以及是否发放奖品，只有在后台服务器做出判定之后，用户本次中奖才生效。

第28行~第42行定义中奖结果的显示情况，当用户刮了10下（以程序得到响应的次数为准）后，如果用户本次中奖，则将涂层区显示成奖项，并且将结果区显示出来，而如果用户本次没有获奖，则在涂层区显示“谢谢参与”。了避免用户在刮完以后继续刮时又弹出结果，使用一个变量display作为显示开关，让结果只显示一次。上述效果如图9-13所示。

图9-13　刮刮卡结果页面

9.6　地图导航

随着交通及经济的发展，人们的活动区域越来越大，不认识路、找不到目的地的情况也屡有发生，地图导航功能可以帮助用户导航到目的地。本节介绍如何在微信中使用地理位置消息来实现导航。

9.6.1　高德地图

高德是中国领先的数字地图内容、导航和位置服务解决方案提供商。高德拥有中国领先的电子地图数据库，并拥有多个“甲级资质”，在汽车导航、政府和企业应用、互联网及移动互联网位置服务应用三大业务领域均处于市场领先地位。

高德地图API是一组基于云的地图服务接口，包括互联网地图API和手机地图API。高德地图API通过互联网、移动互联网向桌面和移动终端用户提供丰富的地图服务功能，如地图显示、标注、位置检索、驾车出行方案检索、公交查询、地址解析等。通过高德地图API，用户可以轻松地在自己的应用中定制强大、快速、轻便的地图功能。基于高德地图云服务，用户无需考虑系统维护，无需购买地图数据，便可以结合业务需求快速构建地图应用，大大降低地图服务的使用成本。目前，新浪、赶集网、搜房网、爱帮网等3万多家知名网站一致调用高德地图API来支持其互联网地图位置业务，同时，超过12万家第三方开发者调用高德地图API服务进行应用开发。

为方便企业和个人开发者使用高德地图服务，高德建立了地图API频道http://api.amap.com/，该频道提供各种开发文档、参考手册、功能演示和工具资源，供开发者下载使用。

我们使用高德地图API完成导航功能。这需要先获取目的地和起始地的坐标，然后根据两地坐标来进行导航。

我们使用高德地图JavaScript API中的“鼠标拾取地图坐标”功能来获得目的坐标。该功能的地址为http://code.autonavi.com/javascript/example/num/0803。

获取地图的示例如图9-14所示。

图9-14　鼠标拾取地图坐标

在图9-14中，当单击地图中的“飞亚达科技大厦”之后，底部显示该处的坐标，纬度为Y值：22.539394，经度为X值：113.956246。该坐标为GCJ-02坐标系的坐标。

高德地图API提供了线路规划功能，给出起点及终点，就可以搜索公交或驾车线路。其接口示例如下所示：

```
http://mo.amap.com/?from=31.234527,121.287689(起点名称)&to=31.234527,121.287689(终点名称)&type=0&opt=1&dev=1。
```

该接口的参数说明如表9-2所示。

表9-2　高德地图线路规划API参数说明

参数	是否必填	说明
from	是	起点坐标，括号内为用户自定义起点名称
to	是	终点坐标，括号内为用户自定义终点名称
type	是	出行方式，type=0（驾车），type=1（公交）
opt	否	驾车：opt=0（速度最快），opt=1（费用最少），opt=2（线路最短），opt=3（不走快速路） 公交：opt=0（速度最快），opt=1（换乘最少），opt=2（步行最少），opt=3（舒适优先）
dev	否	是否已偏移；dev=0（坐标已偏移，默认值），dev=1（坐标未偏移）

9.6.2　开发实现

根据上述接口，可以实现当微信公众账号接收到用户的地理位置消息时，同时返回公交及自驾线路给用户。相关代码如下所示：

```
private function receiveLocation($object)
{
    $Pondbay = array("name" => "方倍工作室", "latitude" => "22.539394", "longitude" => "113.956246");

    $content[] = array("Title" =>"高德地图为您导航",
                        "Description" =>"",
                        "PicUrl" =>"",
                        "Url" =>"");
    $content[] = array("Title" =>"点击图片查看公交线路导航",
                        "Description" =>"",
                        "PicUrl" =>"http://h.hiphotos.bdimg.com/wisegame/pic/item/1fd98d1001e9390186c2c97479ec54e737d196bc.jpg",
                        "Url" =>"http://mo.amap.com/?from=".$object->Location_X.",".$object->Location_Y."(".$object->Label.")&to=".$Pondbay['latitude'].",".$Pondbay['longitu
    $content[] = array("Title" =>"点击图片查看驾车线路导航",
                        "Description" =>"",
                        "PicUrl" =>"http://b.hiphotos.bdimg.com/wisegame/pic/item/eeb1cb1349540923f48a42079058d109b2de49e3.jpg",
                        "Url" =>"http://mo.amap.com/?from=".$object-> Location_X.",".$object->Location_Y."(".$object->Label.")&to=".$Pondbay['latitude'].",".$Pondbay['longit
    $result = $this->transmitNews($object, $content);
    return $result;
}
```

上述代码运行时将返回一个多图文消息，分别显示标题、公交导航、驾车导航的消息。公交和驾车的图文消息的链接为高德地图的接口URL，接口中起始地址为用户发送的地理位置，目的地址为已定义的坐标地址。程序运行后，界面如图9-15所示。

13:46
PondBay
下午13:33
大灰狼餐厅(华侨城店)
广东省深圳市南山区华侨城生态广场A107
高德地图为您导航
单击图片查看公交线路导航
单击图片查看驾车线路导航

图9-15　接收地理位置信息

单击公交线路和驾车线路导航后，高德地图生成的线路规划如图9-16所示。

图9-16　公交线路和驾车线路规划

9.7　智能问答

前面我们使用的都是基于固定查询指令的回复，这样的好处是内容格式统一，方便软件开发人员编写程序做出分析，回复精准的内容给用户。但在生活中，人们问的内容很随意，甚至千差万别，这时，回复内容要想和用户的问题相匹配，就需要更智能的程序。本节介绍如何在微信公众平台上开发更智能的问答系统。

9.7.1　关键词回复

在微信公众平台的高级功能中除了开发模式外，还有编辑模式。编辑模式中有一个关键词自动回复，它是指当用户输入了某个关键词的时候，就触发相应的回复。我们在开发模式中也能实现关键词自动回复。

我们需要先定义一组数组，数组中键为关键词，值为对应的回复，当用户输入的文字能匹配到某个关键词时，则回复该关键词对应的回复内容，我们定义“电话”、“地址”、“微信”三个关键词，将其作为数组中的键，然后为它们定义相应的值，当用户发送的内容中包含相应的关键词名称时，则回复对应的内容。核心代码如下所示：

```
private function receiveText($object)
{
    $content = trim($object->Content);

    $keywords = array(
      "电话" => "0755-87654321",
      "地址" => "广东省深圳市南山区高新南一道飞亚达科技大厦(518057)",
      "微信" => "pondbaystudio"
    );
    foreach ($keywords as $key=>$value){
        $contain = strchr($content, $key);
        if (!empty($contain)){
            $reply = $value;
            break;
        }
    }

    $result = $this->transmitText($object, $reply);
    return $result;
}
```

上述代码最终的运行效果如图9-17所示。

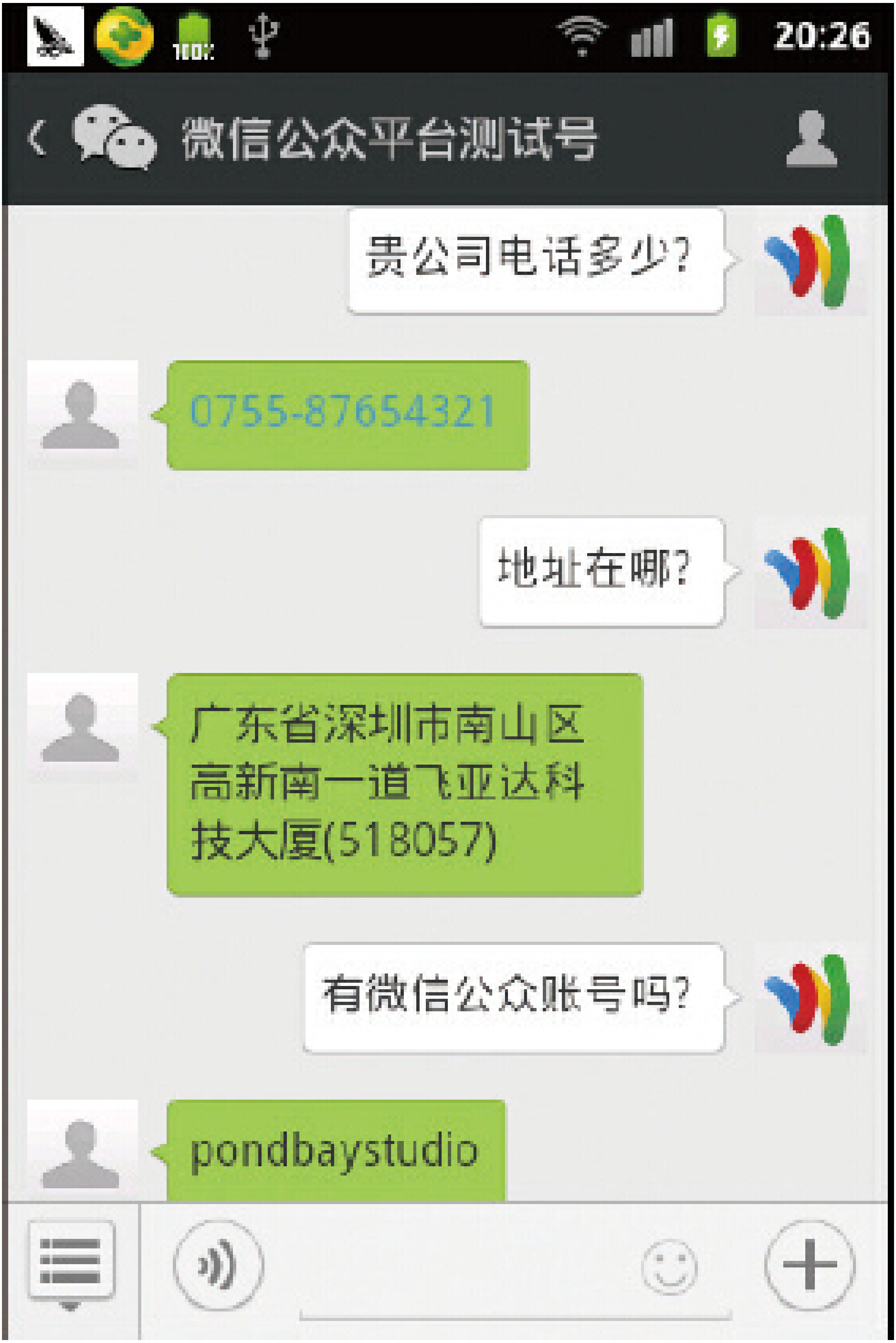

20:26
微信公众平台测试号
贵公司电话多少?
0755-87654321
地址在哪?
广东省深圳市南山区高新南一道飞亚达科技大厦(518057)
有微信公众账号吗?
pondbaystudio

图9-17　关键词自动回复

9.7.2　中文分词

除了关键词回复之外，还可以使用中文分词来帮助我们实现智能回复。

中文分词（Chinese word segmentation）指的是将一个汉字序列切分成一个个单独的词。分词就是将连续的字序列按照一定的规范重新组合成词序列的过程。我们知道，在英文的行文中，单词之间是以空格作为自然分界符的，而中文只是字、句和段能通过明显的分界符来简单划界，唯独词没有一个形式上的分界符，虽然英文也同样存在短语的划分问题，不过在词这一层上，中文比英文要复杂得多、困难得多。中文分词就是要解决句子中词的划分问题，然后可以提取其中的关键词语进行搜索。

打个比方：“深圳天气怎么样”这句话可分词为“深圳”、“天气”、“怎么样”三部分，因为它们代表不同的词语类别及意义。

新浪SAE提供了中文分词服务saeSegment，上述句子进行中文分词的代码如下所示：

```
$str = "深圳天气怎么样";
$seg = newSaeSegment();
$ret = $seg->segment($str, 1);

print_r($ret);
```

代码运行后，结果如下所示：

```
Array
(
    [0] => Array
        (
            [word] =>深圳
            [word_tag] => 102
            [index] => 0
        )
   [1] => Array
        (
            [word] =>天气
            [word_tag] => 95
            [index] => 1
        )
   [2] => Array
        (
            [word] =>怎么样
            [word_tag] => 40
            [index] => 2
        )
)
```

上述结果中，word表示词，word_tag表示词性（102代表地点名词，95代表名词），index表示词在句子中的位置索引。SAE分词词性说明可以参考附录C。

我们把分词功能写成函数，并且将名词作为分类标识、地点作为关键词组合成数组，相关代码如下所示：

```
function segment($str)
{
    $seg = new SaeSegment();
    $ret = $seg->segment($str, 1);

    if ($ret === false){
        return;
    }
    $category = "";
    $keyword = "";
    foreach ($ret as $key => $value) {
        if ($value["word_tag"] == 95){
            $category = $value["word"];
        }
        if ($value["word_tag"] == 102){
            $keyword = $value["word"];
        }
    }
    if (!empty($category) && !empty($keyword)){
        return array('category'=>$category, 'keyword'=>$keyword);
    }else{
        return;
    }
}
```

根据获得的功能类别及关键字，我们可以做出根据地点来查询其相关信息的自动回复，比如天气预报和空气质量。在前面的章节中，我们还提到了将语音识别的结果并入文本消息结果的查询，基于这些，我们可以开发出非常智能的语音识别搜索及文本搜索的功能，相关代码如下所示。

```
private function receiveText($object)
{
    if (isset($object->Recognition)){
        $content = strval($object->Recognition);
        $mediaid = trim($object->MediaID);
    }else{
        $content = trim($object->Content);
    }
    if (isset($_SERVER['HTTP_APPNAME'])){
        include("segment.php");
        $result = sinasegment($content);
        if (is_array($result)){
            switch ($result['category'])
            {
                case "天气":
                    $url = "http://api100.duapp.com/weather/?appkey= book97&city=".urlencode($result['keyword']);
                    $output = file_get_contents($url);
                    $replyContent = json_decode($output, true);
                    break;
                case "空气":
                    $url = "http://api100.duapp.com/airquality/?appkey= book97&city=".urlencode($result['keyword']);
                    $output = file_get_contents($url);
                    $replyContent = json_decode($output, true);
                    break;
                default:
                    $replyContent = "还不支持这一功能：".$result['category'];
                    break;
            }
        }else{
            $replyContent = "不能理解你的内容：".$content;
        }
    }else{
        $replyContent = "只能运行在SAE环境！";
    }

    if (is_array($replyContent)){
```

```
        $resultStr = $this->transmitNews($object, $replyContent);
    }else{
        $resultStr = $this->transmitText($object, $replyContent);
    }
    return $resultStr;
}
```

上述代码中既支持文字查询天气预报和空气质量，也支持语音查询。发送内容为“最近深圳天气好不？”的文字和“北京的空气质量怎么样”的语音后，运行结果如图9-18所示。

图9-18　文本查询和语音查询

9.8　在线客服

在线客服是一种以网络为媒介，向互联网客户与内部员工提供即时沟通的通信技术。在线客服可以增加销售机会，降低运营成本，巩固客户关系。微信公众平台高级接口中提供了在线客服接口，本节介绍如何利用这一接口开发基于微信的在线客服。

9.8.1　流程状态

我们以一个已经初步实现的在线客服系统为原型来讲解实现的步骤。

首先，要定义一个入口，即用户在什么情况下开始和后台的客服员工进行在线沟通。这里在自定义菜单中预置一个二级菜单，菜单名称为“在线客服”，界面如图9-19所示。

图9-19　“在线客服”菜单

当用户单击“在线客服”菜单以后，将启动在线客服系统。后台检索出一位当前处于空闲状态的客服的微信id，系统把用户的微信id和客服的微信id作为一组，生成一个初始的会话关系，并将该关系缓存起来。然后系统使用客服接口分别向客服和用户发出通知消息，要求客服立即回应。

客服收到消息时，可从缓存中查找会话关系，得到用户微信id，当客服向微信公众平台回复内容后，微信公众平台将会话内容转发给用户，会话关系得到确认，这样就在用户和客服之间建立了一个唯一的通道。

客服和用户后续向微信公众平台发送消息时，都是微信公众平台检查双方是否在该通道，然后通过客服接口将问答内容互相转发，实现沟通。

当沟通完毕之后，客服向用户确认是否还有其他问题，没有问题之后，客服也将单击“在线客服”菜单，后台判断是客服单击之后，将删除之前建立的缓存，关闭会话通道，并且同时通知用户和客

服，会话已经结束。

9.8.2 开发实现

为了方便使用客服接口发送消息，先定义微信公众平台高级接口的类，里面包括获取Access Token以及发送客服接口的方法，代码如下所示：

```
class class_weixin
{
    var $appid = "wx3f88af80a4c0a09d";
    var $appsecret = "97b5456aa85b87580406c0013d820e55";

    public function __construct($appid = NULL, $appsecret = NULL)
    {
        if($appid){
            $this->appid = $appid;
        }
        if($appsecret){
            $this->appsecret = $appsecret;
        }

        //hardcode
        $this->lasttime = 1394080550;
        $this->access_token = "x6HnJVWr7YIsl2EOPQAbilEPV1hrAkYHkM0vuqb5v4r3y1I2PcK_iVSaavWYCP0zLvOpqAuYMk0JkYLMhnkYPMBZCRx1qFnoQeHaVUVMRLp254ZXlP0aIs87ytTr5YmvalazENPK-VSPZ6

        if (time() > ($this->lasttime + 7200)){
            $url = "https://api.weixin.qq.com/cgi-bin/token?grant_type= client_credential&appid=".$this->appid."&secret=".$this->appsecret;
            $res = $this->https_request($url);
            $result = json_decode($res, true);
            $this->access_token = $result["access_token"];
        }
    }

    public function send_custom_msg($touser, $type, $data)
    {
        $msg = array('touser' =>$touser);
        switch($type)
        {
            case 'text':
                $msg['msgtype'] = 'text';
                $msg['text']    = array('content'=> $data);
                break;
        }
        $url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$this->access_token;
        return $this->https_request($url, urldecode(json_encode($msg)));
    }

    protected function https_request($url, $data = null)
    {
        $curl = curl_init();
        curl_setopt($curl, CURLOPT_URL, $url);
        curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
        curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
        if (!empty($data)){
            curl_setopt($curl, CURLOPT_POST, 1);
            curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
        }
        curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
        $output = curl_exec($curl);
        curl_close($curl);
        return $output;
    }
}
```

在创建自定义菜单时，将一个菜单的name和key均设置为“在线客服”，该菜单将作为客服的入口。菜单创建时的内容如下所示：

```
{"menu":{"button":[{"name":"关于方倍","sub_button":[{"type":"click","name":"公司简介","key":"公司简介","sub_button":[]},{"type":"click","name":"社会责任","key":"社会责任","sub_butto
```

当用户或客服单击菜单时，相应的代码处理如下所示：

```
private function receiveEvent($object)
{
    $content = "";
    switch ($object->Event)
    {
        case "subscribe":
            $content = "欢迎关注方倍工作室";
        case "unsubscribe":
            break;
        case "CLICK":
            switch ($object->EventKey)
            {
                case "在线客服":
                    $staffs = array("oQW8FuFKL0530Qhp3rh0HciHkTL0");
                    $mmc=memcache_init();
                    require_once('weixin.class.php');
                    $weixin = new class_weixin();

                    //客服单击，结束会话
                    if (in_array($object->FromUserName, $staffs)) {
                        $relation = json_decode(memcache_get($mmc, "service"), true);
                        $from = strval($object->FromUserName);
                        $to = $relation['from'];
                        $weixin->send_custom_msg($from, "text", "你已结束本次对话！");
                        $weixin->send_custom_msg($to, "text", "感谢您的咨询。期待下次再会！");
                        memcache_delete($mmc,"service");
                    }

                    //用户单击，开始会话
                    else {
                        $from = strval($object->FromUserName);
                        $to = $staffs[0];
                        memcache_set($mmc, "service", '{"from":"'.$from. '","to":"'.$to.'"}');
                        $weixin->send_custom_msg($from, "text", "正在为您接入客服，请稍候http://www.hzcourse.com/resource/readBook?path=/openresources/teach_ebook/uncompressed/
                        $weixin->send_custom_msg($to, "text", "有用户请求接入，请响应！");
                    }
                    return;
                default:
                    $content = "菜单默认回复";
                    break;
            }
            break;
        default:
            break;

    }
    if (is_array($content)){
        $result = $this->transmitNews($object, $content);
    }else{
        $result = $this->transmitText($object, $content);
    }
    return $result;
}
```

而当用户或客服发送消息时，消息中转的代码如下所示：

```
private function receiveText($object)
{
    $keyword = trim($object->Content);
    $mmc = memcache_init();
    $service = memcache_get($mmc, "service");

    //当前有人工客服对话
    if ($service){
        $relation = json_decode($service, true);
        //用户处在人工客服交互中
        if (in_array($object->FromUserName, $relation)){
            require_once('weixin.class.php');
            $to = ($relation['from'] == strval($object->FromUserName))?$relation['to']:$relation['from'];
            $weixin = new class_weixin();
            $weixin->send_custom_msg($to, "text", $keyword);
            return;
        }
    }

    $content = "";
    if (is_array($content)){
        $result = $this->transmitNews($object, $content);
    }else{
        $result = $this->transmitText($object, $content);
    }
    return $result;
}
```

当用户单击“在线客服”菜单之后，用户和客服的微信界面效果图如图9-20所示。

图9-20　用户单击菜单请求客服

用户和客服在会话过程中的界面如图9-21所示。

图9-21　用户和客服会话过程中

客服单击“在线客服”菜单关闭会话过程后，用户和客服的界面如图9-22所示。

图9-22　客服单击菜单结束会话

本系统中的在线客服比较简单，但已初具雏形。在现实生活中，客服系统是多用户、多客服的多对多系统，这样的系统需要包括客服员工管理、用户管理以及会话管理，要考虑的核心则是系统资源的调度与分配。

由于微信的特点，也可以添加更多其他的功能，比如客服不在线或全忙时留言回复管理、多消息类型回复等，以此充实和完善客服系统的功能。

9.9　广告效果统计

9.9.1　场景二维码

越来越多的企业在投放多个广告之后，常常面临这样的问题，即不知道每个广告的效果怎么样，也不知道这些广告的转化率怎么样。这是因为没有对广告结果进行统计分析。本节介绍如何利用微信公众平台的场景二维码功能开发广告效果统计功能。

开发思路是预先生成多个场景的二维码，把各二维码放到不同的广告投放地，广告或活动内容引导用户关注扫描后，后台就可以统计出该地点的用户关注数、关注时间等信息。

以下代码批量生成了8个二维码，并将其保存在本地。

```
$appid = "wx3f88af80a4c0a09d";
$appsecret = "97b5456aa85b87580406c0013d820e55";
$url = "https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=$appid&secret=$appsecret";

$output = https_request($url);
$jsoninfo = json_decode($output, true);
$access_token = $jsoninfo["access_token"];

//永久二维码
for ($i = 1; $i <= 8; $i++) {
    $scene_id = $i;
    $qrcode = '{"action_name": "QR_LIMIT_SCENE", "action_info": {"scene": {"scene_id": '.$scene_id.'}}}';

    $url = "https://api.weixin.qq.com/cgi-bin/qrcode/create?access_token=$access_token";
    $result = https_request($url, $qrcode);
    $jsoninfo = json_decode($result, true);
    $ticket = $jsoninfo["ticket"];
    $url = "https://mp.weixin.qq.com/cgi-bin/showqrcode?ticket=".urlencode($ticket);
    $imageInfo = downloadWeixinFile($url);

    $filename = "qrcode".$scene_id.".jpg";
    $local_file = fopen($filename, 'w');
    fwrite($local_file, $imageInfo["body"]);
    fclose($local_file);
}

//http请求
function https_request($url, $data = null)
```

```
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}

//下载文件
function downloadWeixinFile($url)
{
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_NOBODY, 0);    //只取body头
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $package = curl_exec($ch);
    $httpinfo = curl_getinfo($ch);
    curl_close($ch);
    $imageAll = array_merge(array('body' => $package), array('header' => $httpinfo));
    return $imageAll;
}
```

生成二维码后，如图9-23所示。

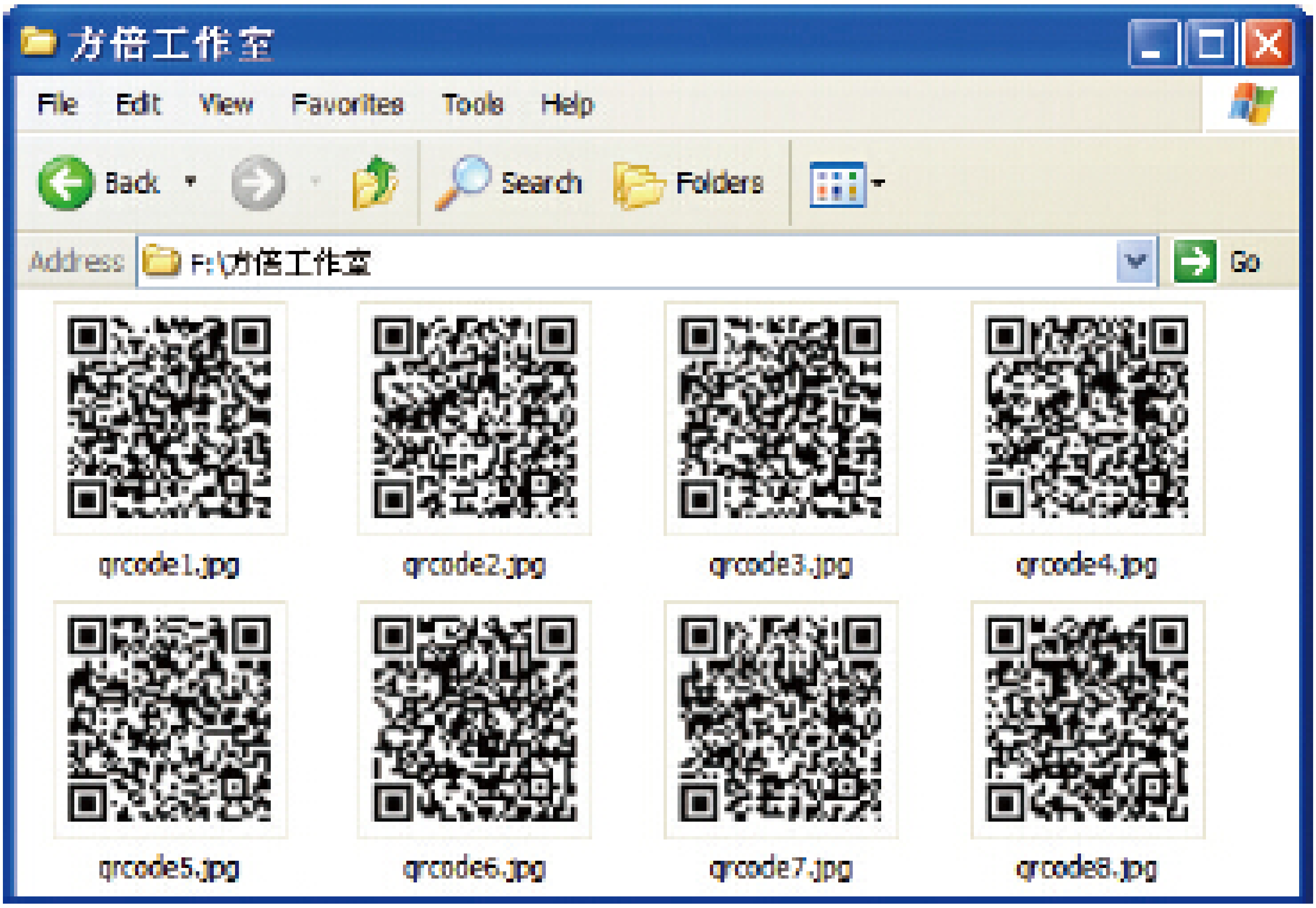


图9-23　8个永久场景二维码

将8个场景二维码放到8个不同的广告投放地，并且引导用户关注，这样才能获得用户关注的相关数据。

9.9　广告效果统计

9.9.1　场景二维码

越来越多的企业在投放多个广告之后，常常面临这样的问题，即不知道每个广告的效果怎么样，也不知道这些广告的转化率怎么样。这是因为没有对广告结果进行统计分析。本节介绍如何利用微信公众平台的场景二维码功能开发广告效果统计功能。

开发思路是预先生成多个场景的二维码，把各二维码放到不同的广告投放地，广告或活动内容引导用户关注扫描后，后台就可以统计出该地点的用户关注数、关注时间等信息。

以下代码批量生成了8个二维码，并将其保存在本地。

```
$appid = "wx3f88af80a4c0a09d";
$appsecret = "97b5456aa85b87580406c0013d820e55";
$url = "https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=$appid&secret=$appsecret";

```

```
$output = https_request($url);
$jsoninfo = json_decode($output, true);
$access_token = $jsoninfo["access_token"];

//永久二维码
for ($i = 1; $i <= 8; $i++) {
    $scene_id = $i;
    $qrcode = '{"action_name": "QR_LIMIT_SCENE", "action_info": {"scene": {"scene_id": '.$scene_id.'}}}';

    $url = "https://api.weixin.qq.com/cgi-bin/qrcode/create?access_token=$access_token";
    $result = https_request($url, $qrcode);
    $jsoninfo = json_decode($result, true);
    $ticket = $jsoninfo["ticket"];
    $url = "https://mp.weixin.qq.com/cgi-bin/showqrcode?ticket=".urlencode($ticket);
    $imageInfo = downloadWeixinFile($url);

    $filename = "qrcode".$scene_id.".jpg";
    $local_file = fopen($filename, 'w');
    fwrite($local_file, $imageInfo["body"]);
    fclose($local_file);
}

//http请求
function https_request($url, $data = null)
{
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
    if (!empty($data)){
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
    }
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    $output = curl_exec($curl);
    curl_close($curl);
    return $output;
}

//下载文件
function downloadWeixinFile($url)
{
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_NOBODY, 0);    //只取body头
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, FALSE);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    $package = curl_exec($ch);
    $httpinfo = curl_getinfo($ch);
    curl_close($ch);
    $imageAll = array_merge(array('body' => $package), array('header' => $httpinfo));
    return $imageAll;
}
```

生成二维码后，如图9-23所示。

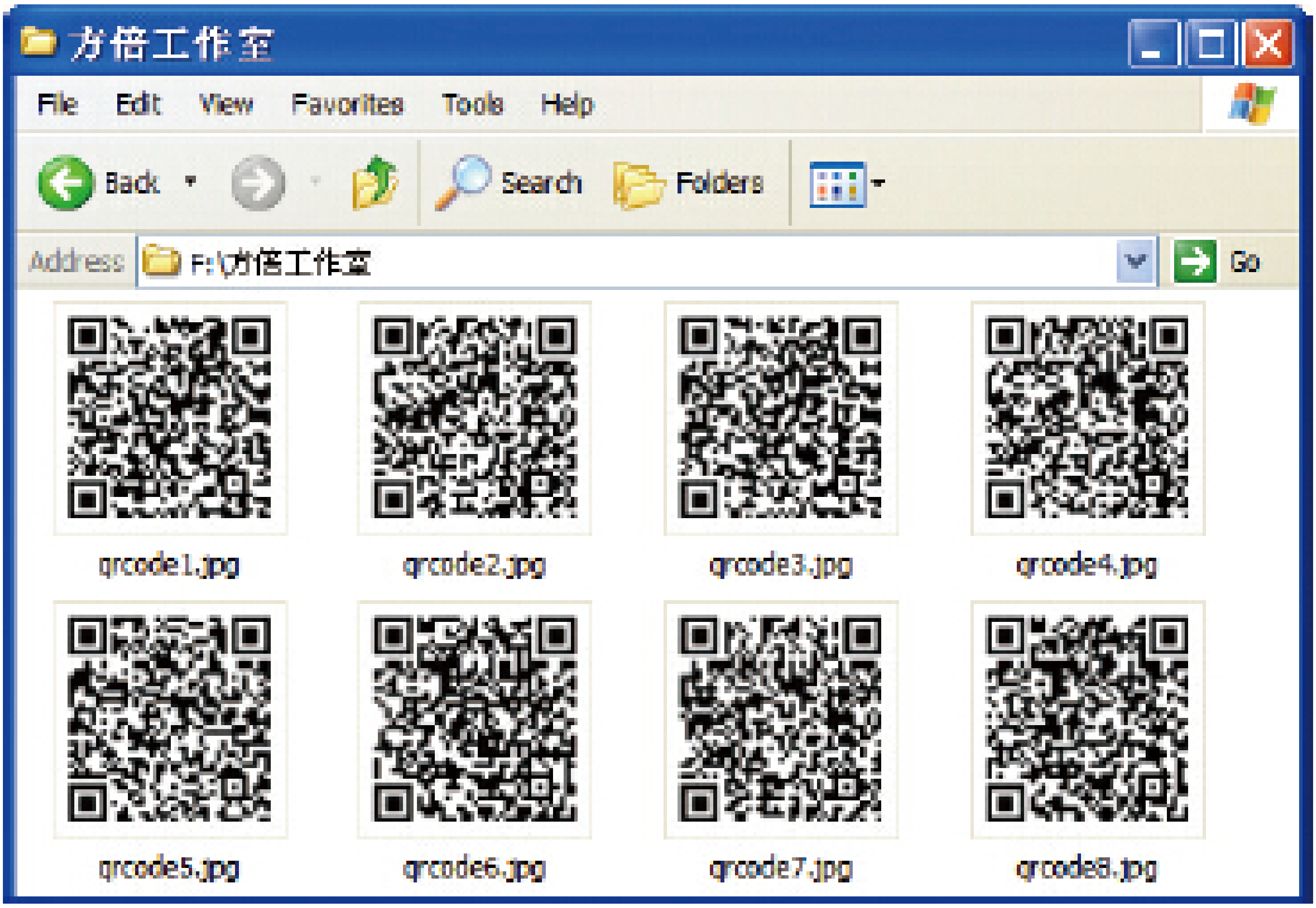


图9-23　8个永久场景二维码

将8个场景二维码放到8个不同的广告投放地，并且引导用户关注，这样才能获得用户关注的相关数据。

9.9.2　数据统计

我们需要设计一个表，用于存储统计记录，SQL脚本如下所示：

```
DROPTABLEIFEXISTS 'qrcode';
CREATETABLEIFNOTEXISTS 'qrcode' (
  'id' int(16) NOTNULL auto_increment,
  'scene' varchar(2) NOTNULL COMMENT '场景',
  'year' varchar(4) NOTNULL COMMENT '年',
  'month' varchar(2) NOTNULL COMMENT '月',
  'day' varchar(2) NOTNULL COMMENT '日',
PRIMARYKEY  ('id')
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=0 ;
```

上述脚本定义了5个字段，用于存储用户关注时的场景id以及关注时的年、月、日。另外定义了主键id。

定义数据库配置文件如下：

```
if (isset($_SERVER['HTTP_APPNAME'])){ //SAE
    define("MYSQLHOST", SAE_MYSQL_HOST_M);
    define("MYSQLPORT", SAE_MYSQL_PORT);
    define("MYSQLUSER", SAE_MYSQL_USER);
    define("MYSQLPASSWORD", SAE_MYSQL_PASS);
    define("MYSQLDATABASE", SAE_MYSQL_DB);  //
}else{
    define("MYSQLHOST", "localhost");
    define("MYSQLPORT", "3306");
    define("MYSQLUSER", "root");
    define("MYSQLPASSWORD", "root");
    define("MYSQLDATABASE", "weixin");  //
}
```

定义mysql类如下：

```
class class_mysql
{
    function __construct(){
        $host = MYSQLHOST;
        $port = MYSQLPORT;
        $user = MYSQLUSER;
        $pwd =  MYSQLPASSWORD;
        $dbname = MYSQLDATABASE;

        $link = @mysql_connect("{$host}:{$port}", $user, $pwd, true);
        mysql_select_db($dbname, $link);
        return $link;
    }

    //返回数组
    function query_array($sql){
        $result = mysql_query($sql);
        if(!$result)return false;
        $arr = array();
        while ($row = mysql_fetch_assoc($result)) {
            $arr[] = $row;
        }
        return $arr;
    }

    //只执行
    function query($sql)
    {
        if (!($query = mysql_query($sql))){
            return false;
        }
        return $query;
    }
}
```

当用户关注二维码的时候，后台根据关注时间消息，将该二维码的值写入数据库中。事件响应部分代码如下所示：

```
private function receiveEvent($object)
{
    $content = "";
    switch ($object->Event)
    {
        case "subscribe":
            $content = "欢迎关注方倍工作室";
            if (isset($object->EventKey)){
                $sceneid = str_replace("qrscene_","",$object->EventKey);
                require_once('mysql.php');
                $db = new class_mysql();
                $sql = "INSERT INTO `qrcode` (`id`, `scene`, `year`, `month`, `day`) VALUES (NULL, '".$sceneid."', '".date("Y")."', '".date("m")."', '".date("d")."')";
                $db->query($sql);
            }
            break;
        case "unsubscribe":
            $content = "取消关注";
            break;
        case "SCAN":
            $content = "扫描场景 ".$object->EventKey;
            break;
        default:
            break;
    }
    if (is_array($content)){
        $result = $this->transmitNews($object, $content);
    }else{
        $result = $this->transmitText($object, $content);
    }
    return $result;
}
```

这样，后台就能统计到用户关注的数据了。

mysql中最终存储的数据如图9-24所示。

←T→			id	scene 场景	year 年	month 月	day 日
			1	2	2014	01	01
			2	2	2014	03	01
			3	3	2014	03	01
			4	4	2014	03	01
			5	2	2014	03	03
			6	1	2014	05	03
			7	3	2014	06	03
			8	2	2014	08	03
			9	2	2015	08	02
			10	3	2014	02	14
			11	1	2014	03	10
			12	8	2014	03	10
			13	2	2014	03	10

图9-24　关注统计数据

我们使用SQL语句来获取统计数据。

比如统计2014年3月各场景关注情况的SQL查询脚本如下所示。

```
SELECTCOUNT( scene ) , scene
FROM 'qrcode'
WHEREYEAR='2014'
ANDMONTH='03'
GROUPBY scene
```

获取到的结果数据如图9-25所示。

COUNT(scene)	scene 场景
140	1
110	2
77	3
104	4
29	5
161	6
13	7
195	8

图9-25　3月份各场景统计数据

我们使用JpGraph来生成统计图。

JpGraph是PHP下一个面向对象的图表创建库，你只需从数据库中取出相关数据，定义标题、图表类型，就能轻松画出折线图、柱形图、饼状图等图表。官方网站为http://jpgraph.net/，可下载其使用手册了解使用方法。

下面代码使用3月份的场景统计数据创建了一个柱状统计图。

```
require_once ('jpgraph/jpgraph.php');
require_once ('jpgraph/jpgraph_bar.php');
require_once ('jpgraph/jpgraph_line.php');

// 数据
$data_follow = array(140,110,77,104,29,161,13,195);

// 构造对象
$graph = new Graph(320,440);          // 屏幕分辨率

// 基本参数
$graph->SetScale("textlin");               // 线性标尺
$graph->SetY2Scale('lin',0,100);  // 对数
$graph->Set90AndMargin(50,0,65,0);         // 旋转90度
$graph->yaxis->SetTitleMargin(25);

// 标题与字体
$graph->title->Set("Scene Analysis");
$graph->title->SetFont(FF_FONT1,FS_BOLD);
$graph->xaxis->title->Set("Sce");
$graph->yaxis->title->Set("Num");
$graph->y2axis->SetColor('black','blue');
$graph->y2axis->SetLabelFormat('%2d');

// 生成柱状图
$bplot = newBarPlot($data_follow);
$bplot->SetFillColor("orange@0.2");
$bplot->SetValuePos('center');
$bplot->value->SetFormat("%d");
$bplot->value->SetFont(FF_ARIAL,FS_NORMAL,9);
$bplot->value->Show();

// 柱状图叠到图形中
$graph->Add($bplot);

// 生成图形
return$graph->Stroke();
```

其运行结果在微信中显示如图9-26所示。

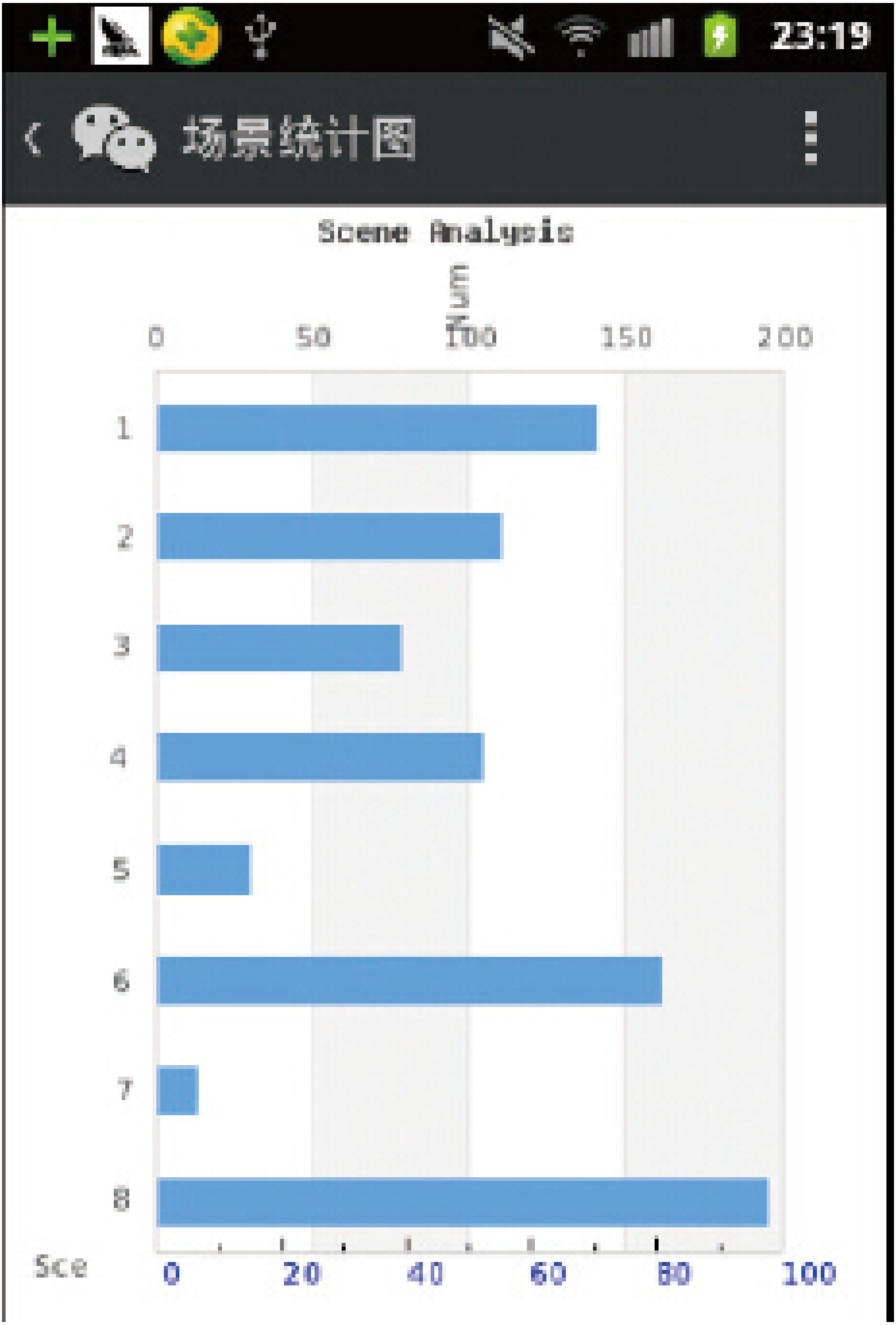


图9-26　3月份各场景统计柱状图

9.10　本章小结

本章介绍了会员卡、预约订单、产品相册、大转盘、刮刮卡、地图导航、智能问答、在线客服、广告效果统计等功能的实现，以期帮助企业实现更好的营销功能及提供更好的服务。

第10章　微信开发实用技巧

在微信公众平台的开发中，除了基于微信公众平台接口以及常规的Web开发之外，还有一些并未公开的功能。这些功能，或者能在某些特定的方面丰富我们的功能，或者可定制想要的内容。本章介绍这些不常见但却比较实用的小技巧。

10.1　实现表情飘落效果

表情飘落是指文字内容中出现某些词语的时候，微信窗口中将从上至下飘落跟词语内容相关的表情。图10-1是当内容中包含“生日快乐”这一词语的时候，飘落生日蛋糕的场景。无论是用户发送，还是公众账号回复，都能产生该效果。

图10-1　表情飘落

这些表情中，有的还需要在特定节日期间才出现。目前已知的可用触发词及其飘落的表情如表10-1所示。

表10-1　表情飘落触发词列表

表情	期限	触发词
蛋糕	长期	birthday, 生日快乐
吻	长期	xoxo，么么哒，cium，baci，besos
星星	长期	miss u，想你了
心	长期	点个赞
火苗	春节	红红火火
爆竹	春节	新年快乐
黄色的星星	春节	吉星高照，福星高照

（续）

表情	期限	触发词
桔子	春节	大吉大利，吉祥如意，万事如意，万事大吉
鱼	春节	年年有余，年年有鱼
苹果	春节	一路平安，平平安安
钱袋	春节	恭喜发财，招财进宝
马头	2014 年春节	马年快乐，马到成功，马到功成，馬年進步，馬年快樂
圣诞树	圣诞节期间	圣诞快乐
雪花	《来自星星的你》播出后	炸鸡和啤酒

使用这一功能，我们可以在某些自动回复中添加相应的触发词，增加互动气氛。

需要说明的是，这些触发词并不是一直有效，也不是在所有微信版本中有效，而是会一直不停地变动。由于这不是官方公布的信息，所以获取触发词列表需要自己多尝试以及关注相关方面的资讯。

10.2　回复QQ表情和Emoji表情

微信的文字内容发送框里面有一个笑脸符号，单击该笑脸后，可以看到下面有很多表情图标。在使用微信发送消息时，我们可以发送表情。这些表情包括QQ表情和Emoji表情（也叫符号表情），如图10-2所示。

图10-2　微信表情

用户和公众账号都能直接发送或回复相应的QQ表情给对方。微信公众账号回复QQ表情的时候，需要回复相应的表情代码，这样就能发送出相应的表情。这些表情代码被当作文本消息在微信中转义成图标。QQ表情的相应代码如表10-2所示。

表10-2　QQ表情代码列表

代码	表情	代码	表情	代码	表情	代码	表情
/:&>	右太极	/::*	亲亲	/:hug	拥抱	/:|-)	困
/:<&	左太极	/:X-)	阴险	/:gift	礼物	/::g	饥饿
/:kiss	献吻	/:P-(	快哭了	/:sun	太阳	/:,@o	傲慢
/:hiphot	乱舞	/:>-|	委屈	/:moon	月亮	/::d	白眼
/:#-0	激动	/::-O	鄙视	/:shit	便便	/:,@-D	愉快
/:oY	头像	/:@>	右哼哼	/:ladybug	瓢虫	/:,@P	偷笑
/:skip	跳绳	/:<@	左哼哼	/:footb	足球	/::T	吐
/:turn	回头	/:B-)	坏笑	/:kn	刀	/::Q	抓狂
/:kotow	磕头	/:&-(	糗大了	/:bome	炸弹	/:--b	冷汗
/:circle	转圈	/:handclap	鼓掌	/:li	闪电	/::+	酷
/:<O>	怄火	/:dig	抠鼻	/:cake	蛋糕	/::(	难过
/:shake	发抖	/:wipe	擦汗	/:break	心碎	/::O	惊讶
/:jump	跳跳	/:bye	再见	/:showlove	爱心	/::D	呲牙
/:<L>	飞吻	/:xx	敲打	/:heart	嘴唇	/::P	调皮
/:love	爱情	/:!!!	骷髅	/:fade	凋谢	/::@	发怒
/:ok	OK	/:,@!	衰	/:rose	玫瑰	/::-|	尴尬
/:no	NO	/::8	疯了	/:pig	猪头	/::-'(	大哭
/:lvu	爱你	/:,@@	晕	/:eat	饭	/::Z	睡
/:bad	差劲	/:,@x	嘘	/:coffee	咖啡	/::X	闭嘴
/:@@	拳头	/:?	疑问	/:oo	乒乓	/::$	害羞
/:jj	勾引	/::-S	咒骂	/:basketb	篮球	/::<	流泪
/:@)	抱拳	/:,@f	奋斗	/:beer	啤酒	/:8-)	得意
/:v	胜利	/::,@	悠闲	/:<W>	西瓜	/::|	发呆
/:share	握手	/::>	憨笑	/:pd	菜刀	/::B	色
/:weak	弱	/::L	流汗	/:8*	可怜	/::~	撇嘴
/:strong	强	/::!	惊恐	/:@x	吓	/::)	微笑

可以使用如下代码来实现文字到QQ表情的转换，这样，在回复的内容中如果有相应的词语，就能换成相应的QQ表情。

```
$face = array('/::)','/::~','/::B','/::|','/:8-)','/::<','/::$','/::X','/::Z','/::\'(','/::-|','/::@','/::P','/::D','/::O','/::(','/::+','/:Cb','/::Q','/::T','/:,@P','/:,@-D','
$word = array('微笑','伤心','美女','发呆','墨镜','哭','羞','哑','睡','哭','囧','怒','调皮','笑','惊讶','难过','酷','汗','抓狂','吐','笑','快乐','奇','傲','饿','累','吓','汗','高兴','闲',
$content = str_replace($face, $word, $message);
```

Emoji即表情符号，词义来自日语（えもじ，e-moji，moji在日语中的含义是字符），是由栗田穣崇（Shigetaka Kurit）创建的，目前已普遍应用于网络聊天软件中。Emoji表情的一部分如图10-3所示。

图10-3　Emoji表情

Emoji表情包含多种版本，包括Unified、DoCoMo、KDDI、Softbank以及Google，其中Unified是最官方的版本，也是微信中使用的版本。Emoji表情使用十六进制代码标记，其符号达700多个。各符号名称、含义及其在各种版本下的十六进制代码可以参考官方网站，官网地址为：

```
http://www.unicode.org/~scherer/emoji4unicode/
```

各图标的详细说明可以从以下地址下载：

```
http://www.unicode.org/~scherer/emoji4unicode/snapshot/emojidata.pdf
```

也可以在下列网站进行查询：

```
http://code.iamcal.com/php/emoji/
```

在微信公众平台中，公众账号回复Emoji表情，不能直接回复代码，需要进行ASCII转码，然后将编码以文本消息回复。十六进制转码代码如下：

```
function utf8_bytes($cp)
{
    if ($cp > 0x10000){
        # 4 bytes
        return    chr(0xF0 | (($cp & 0x1C0000) >> 18)).
            chr(0x80 | (($cp & 0x3F000) >> 12)).
            chr(0x80 | (($cp & 0xFC0) >> 6)).
            chr(0x80 | ($cp & 0x3F));
    }else if ($cp > 0x800){
        # 3 bytes
        return    chr(0xE0 | (($cp & 0xF000) >> 12)).
            chr(0x80 | (($cp & 0xFC0) >> 6)).
            chr(0x80 | ($cp & 0x3F));
    }else if ($cp > 0x80){
        # 2 bytes
        return    chr(0xC0 | (($cp & 0x7C0) >> 6)).
            chr(0x80 | ($cp & 0x3F));
    }else{
        # 1 byte
        return chr($cp);
    }
}
```

构造文本消息时，将相应的十六进制作为参数填入转码函数中即可，特别要注意的是，部分表情有两组十六进制代码，比如“中国国旗”、“美国国旗”，需要将两组十六进制都进行转码后拼接，代码实现如下所示：

```
private function receiveText($object)
{
    $keyword = trim($object->Content);
    $content = "中国国旗：".utf8_bytes(0x1F1E8).utf8_bytes(0x1F1F3)."\n".
        "美国国旗：".utf8_bytes(0x1F1FA).utf8_bytes(0x1F1F8)."\n".
```

```
        "男女牵手：".utf8_bytes(0x1F46B)."\n".
        "仙人掌：".utf8_bytes(0x1F335)."\n".
        "电话机：".utf8_bytes(0x260E)."\n".
        "药丸：".utf8_bytes(0x1F48A);
    $result = $this->transmitText($object, $content);
    return $result;
}
```

运行后，效果如图10-4所示。

图10-4　微信公众账号回复Emoji表情

10.3　获取微信版本及手机系统

HTTP_USER_AGENT是用来检查浏览页面的访问者在用什么操作系统（包括版本号）、浏览器（包括版本号）和用户个人偏好的代码。

通过获取微信内置浏览器的User Agent，可以得到用户手机情况及微信版本信息。

经过测试，在iPhone手机下，User Agent类似如下：

```
Mozilla/5.0 (iPhone; CPU iPhone OS 5_1 like Mac OS X) AppleWebKit/534.46 (KHTML, like Gecko) Mobile/9B176 MicroMessenger/5.0.1
```

在Android手机下，UserAgent返回类似如下：

```
Mozilla/5.0 (Linux; U; Android 2.3.6; zh-cn; GT-S5660 Build/GINGERBREAD) AppleWebKit/533.1 (KHTML, like Gecko) Version/4.0 Mobile Safari/533.1 MicroMessenger/4.5.255
```

由此可知，微信浏览器的关键字为MicroMessenger，其后面的数字代表当前的微信版本号，通过识别是否有iPhone以及Android字段以及MicroMessenger及其后面的数号可以获取微信版本及手机型号。

实现代码如下所示：

```
<?php
$ua = $_SERVER['HTTP_USER_AGENT'];
if(!strpos($ua, 'MicroMessenger')){
    $weixin = "不是微信浏览器";
}else{
    $preg = "/MicroMessenger\/(.+)/";
    preg_match_all($preg, $ua, $new_cnt);
    $weixin = "".$new_cnt[1][0]."\n";
}
if(strpos($ua, 'Android')){
    $phone = "Android";
}else if(strpos($ua, 'iPhone OS')){
    $phone = "iOS";
}else{
    $phone = "其他";
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
  <HEAD>
    <TITLE>方倍工作室</TITLE>
    <META charset=utf-8>
     <META name=viewport content="width=device-width, user-scalable=no, initial-scale=1">
    <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.min.css" />
     <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.min.js"></script>
  </HEAD>
  <BODY>
    <div data-role="page" id="page1">
      <div data-role="content">
        <UL data-role="listview" data-inset="true">
          <LI>
            <P>
              <div class="fieldcontain">
                <label for="userid">微信版本</label>
                <input name="userid" id="userid" value="<?php echo $weixin;?>" type="text" >
              </div>
              <div class="fieldcontain">
                <label for="openid">手机系统</label>
                <input name="openid" id="openid" value="<?php echo $phone;?>" type="text" >
              </div>
            </P>
          </LI>
        </UL>
      </div>
      <div data-theme="b" data-role= "footer"
 data-position="fixed">
        <h3>方倍工作室</h3>
      </div>
    </div>
  </BODY>
</HTML>
```

上述代码运行效果如图10-5所示。

18:49
方倍工作室
微信版本
5.2.380
手机系统
Android

图10-5　获得微信版本与手机系统

10.4　微信JS接口使用

10.4.1　绑定JS接口安全域名

使用微信JS接口需要先绑定域名，只有在绑定安全域名下的程序才能调用JS接口。

绑定域名的方法如下。

在微信公众账号“设置”功能下，找到“公众号设置”菜单，如图10-6所示。

图10-6　公众号设置

在“公众号设置”界面切换到“功能设置”界面中，单击右侧的“设置”链接，如图10-7所示。

图10-7　功能设置

在“JS接口安全域名”设置界面中，填写要设置的一级域名，如图10-8所示。

图10-8 JS接口安全域名

填写完成后，界面如图10-9所示。

图10-9 JS接口安全域名已设置

10.4 微信JS接口使用

10.4.1 绑定JS接口安全域名

使用微信JS接口需要先绑定域名，只有在绑定安全域名下的程序才能调用JS接口。

绑定域名的方法如下。

在微信公众账号“设置”功能下，找到“公众号设置”菜单，如图10-6所示。

图10-6　公众号设置

在“公众号设置”界面切换到“功能设置”界面中，单击右侧的“设置”链接，如图10-7所示。

图10-7　功能设置

在“JS接口安全域名”设置界面中，填写要设置的一级域名，如图10-8所示。

图10-8　JS接口安全域名

填写完成后，界面如图10-9所示。

图10-9　JS接口安全域名已设置

10.4.2　微信JS接口SDK

微信JS接口PHP SDK实现如下，注意下述代码无法在SAE中运行，它需要将access_token和jsapi_ticket写入当前目录，并需要有当前目录的写入权限。同时，需要注意的是access_token和jsapi_ticket应该进行全局存储与更新。

```
<?php
class JSSDK {
  private $appId;
  private $appSecret;
  public function __construct($appId, $appSecret) {
    $this->appId = $appId;
    $this->appSecret = $appSecret;
  }
  public function getSignPackage() {
    $jsapiTicket = $this->getJsApiTicket();
    // 注意 URL 一定要动态获取,不能 hardcode.
    $protocol = (!empty($_SERVER['HTTPS']) && $_SERVER['HTTPS'] !== 'off' || $_SERVER['SERVER_PORT'] == 443) ? "https:// " : "http://";
    $url = "$protocol$_SERVER[HTTP_HOST]$_SERVER[REQUEST_URI]";
    $timestamp = time();
    $nonceStr = $this->createNonceStr();
    // 这里参数的顺序要按照 key 值 ASCII 码升序排序
    $string = "jsapi_ticket=$jsapiTicket&noncestr=$nonceStr&timestamp=$timestamp&url=$url";
    $signature = sha1($string);
    $signPackage = array(
      "appId"     => $this->appId,
      "nonceStr"  => $nonceStr,
      "timestamp" => $timestamp,
      "url"       => $url,
      "signature" => $signature,
      "rawString" => $string
    );
    return $signPackage;
  }
  private function createNonceStr($length = 16) {
    $chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
    $str = "";
    for ($i = 0; $i < $length; $i++) {
      $str .= substr($chars, mt_rand(0, strlen($chars) - 1), 1);
    }
    return $str;
  }
  private function getJsApiTicket() {
    // jsapi_ticket 应该全局存储与更新,以下代码以写入到文件中做示例
    $data = json_decode(file_get_contents("jsapi_ticket.json"));
    if ($data->expire_time < time()) {
      $accessToken = $this->getAccessToken();
      // 如果是企业号用以下 URL 获取 ticket
      // $url = "https:// qyapi.weixin.qq.com/cgi-bin/get_jsapi_ticket?access_token=
$accessToken";
      $url = "https:// api.weixin.qq.com/cgi-bin/ticket/getticket?type=jsapi&access
_token=$accessToken";
      $res = json_decode($this->httpGet($url));
      $ticket = $res->ticket;
      if ($ticket) {
        $data->expire_time = time() + 7000;
        $data->jsapi_ticket = $ticket;
        $fp = fopen("jsapi_ticket.json", "w");
        fwrite($fp, json_encode($data));
        fclose($fp);
      }
    } else {
      $ticket = $data->jsapi_ticket;
    }
    return $ticket;
```

```
    }
    private function getAccessToken() {
      // access_token 应该全局存储与更新,以下代码以写入到文件中做示例
      $data = json_decode(file_get_contents("access_token.json"));
      if ($data->expire_time < time()) {
        // 如果是企业号用以下URL获取access_token
        // $url = "https:// qyapi.weixin.qq.com/cgi-bin/gettoken?corpid=$this->appId&corpsecret=
  $this->appSecret";
        $url = "https:// api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=
  $this->appId&secret=$this->appSecret";
        $res = json_decode($this->httpGet($url));
        $access_token = $res->access_token;
        if ($access_token) {
          $data->expire_time = time() + 7000;
          $data->access_token = $access_token;
          $fp = fopen("access_token.json", "w");
          fwrite($fp, json_encode($data));
          fclose($fp);
        }
      } else {
        $access_token = $data->access_token;
      }
      return $access_token;
    }
    private function httpGet($url) {
      $curl = curl_init();
      curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
      curl_setopt($curl, CURLOPT_TIMEOUT, 500);
      curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
      curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, false);
      curl_setopt($curl, CURLOPT_URL, $url);
      $res = curl_exec($curl);
      curl_close($curl);
      return $res;
    }
  }
```

10.4.3 JS文件引入与注入权限验证配置

在使用JS接口之前，需要先引入JS接口文件，同时通过config接口注入权限验证配置，获得签名。

```
<?php
require_once "jssdk.php";
$jssdk = new JSSDK("wx35e39b54516439b6", "85b3e58df5f016aebcd990ee09851b35");
$signPackage = $jssdk->GetSignPackage();
?>
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>微信</title>
  <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=0">
  <link rel="stylesheet" href="http://demo.open.weixin.qq.com/jssdk/css/style.css">
</head>
<body ontouchstart="">
</body>
<script src="http://res.wx.qq.com/open/js/jweixin-1.0.0.js"></script>
<script>
  wx.config({
    debug: false,
    appId: '<?php echo $signPackage["appId"];?>',
    timestamp: <?php echo $signPackage["timestamp"];?>,
    nonceStr: '<?php echo $signPackage["nonceStr"];?>',
    signature: '<?php echo $signPackage["signature"];?>',
    jsApiList: [
      // 所有要调用的 API 都要加到这个列表中
        'checkJsApi',
        'openLocation',
        'getLocation'
      ]
  });
</script>
<script src="demo.js"> </script>
</html>
```

JS支持的接口如下：

（1）基础接口

· 判断当前客户端是否支持指定JS接口checkJsApi

（2）分享接口

· 获取“分享到朋友圈”按钮单击状态及自定义分享内容接口onMenuShareTimeline

· 获取“分享给朋友”按钮单击状态及自定义分享内容接口onMenuShareAppMessage

· 获取“分享到QQ”按钮单击状态及自定义分享内容接口onMenuShareQQ

· 获取“分享到腾讯微博”按钮单击状态及自定义分享内容接口onMenuShareWeibo

（3）图像接口

· 拍照或从手机相册中选图接口chooseImage

· 预览图片接口previewImage

· 上传图片接口uploadImage

· 下载图片接口downloadImage

（4）音频接口

· 开始录音接口startRecord

· 停止录音接口stopRecord

· 播放语音接口playVoice

· 暂停播放接口pauseVoice

· 停止播放接口stopVoice

· 上传语音接口uploadVoice

· 下载语音接口downloadVoice

（5）智能接口

· 识别音频并返回识别结果接口translateVoice

（6）设备信息接口

· 获取网络状态接口getNetworkType

（7）地理位置接口

· 使用微信内置地图查看位置接口openLocation

· 获取地理位置接口getLocation

（8）界面操作接口

· 隐藏右上角菜单接口hideOptionMenu

· 显示右上角菜单接口showOptionMenu

· 关闭当前网页窗口接口closeWindow

· 批量隐藏功能按钮接口hideMenuItems

· 批量显示功能按钮接口showMenuItems

· 隐藏所有非基础按钮接口hideAllNonBaseMenuItem

· 显示所有功能按钮接口showAllNonBaseMenuItem

（9）微信扫一扫

· 调起微信扫一扫接口scanQRCode（微信处理结果）

· scanQRCode（直接返回结果）

（10）微信小店接口

· 跳转微信商品页接口openProductSpecificView

（11）微信卡券接口

· 批量添加卡券接口addCard

· 调起适用于门店的卡券列表并获取用户选择列表chooseCard

· 查看微信卡包中的卡券接口openCard

（12）微信支付接口

· 发起一个微信支付请求chooseWXPay

10.4.4 JS接口相关功能开发

当用户把喜欢的页面发送给朋友或分享到朋友圈时，将弹出一个窗口，该窗口可以自定义标题、内容、图片、链接信息等。好友看到的分享消息由这些内容组成。

发送给朋友与分享到朋友圈的JS脚本开发代码如下所示。

```
wx.ready(function () {
   wx.onMenuShareAppMessage({
       title: '爱情贺卡',
       desc: '山无陵,天地合,乃敢与君绝。',
       link: 'http://www.cnblogs.com/txw1958/',
       imgUrl: 'http://bcs.duapp.com/api100/image/logo/newyear.jpg',
       success: function (res) {
           alert('已分享');
       },
       cancel: function (res) {
           alert('已取消');
       },
       fail: function (res) {
           alert(JSON.stringify(res));
       }
   });
   wx.onMenuShareTimeline({
       title: '新年贺卡',
       link: 'http://www.cnblogs.com/txw1958/',
       imgUrl: 'http://bcs.duapp.com/api100/image/logo/newyear.jpg',
       success: function (res) {
           alert('已分享');
       },
       cancel: function (res) {
           alert('已取消');
       },
       fail: function (res) {
```

```
            alert(JSON.stringify(res));
        }
    });
});
```

onMenuShareAppMessage、onMenuShareTimeline分别是发送给朋友和分享到朋友圈的接口，相关参数如表10-3所示。

表10-3　微信分享接口参数说明

参数	说明	参数	说明
title	标题	link	页面链接
desc	描述	img_url	图片地址

定义后，上述页面执行“分享给好友”和“分享到朋友圈”时的效果如图10-10所示。可以看到，其显示的窗口和我们在代码中定义的一致。

图10-10　发送给好友和分享到朋友圈

在使用接口之前，还可以检测微信客户端版本是否支持指定JS接口，下面是检查是否支持“获取地理位置接口”的代码，如果支持则将用户的坐标带入要跳转的URL的参数中。

```
wx.ready(function () {
    wx.checkJsApi({
        jsApiList: [
            'getLocation'
        ],
        success: function (res) {
            if (res.checkResult.getLocation == false) {
                alert('你的微信版本太低,不支持微信JS接口,请升级到最新的微信版本! ');
                return;
            }
        }
    });
    wx.getLocation({
        success: function (res){
            window.location.href = "location.php?longitude=" + res.longitude + "&latitude=" + res.latitude;
        },
        cancel: function (res) {
            alert('用户拒绝授权获取地理位置');
        }
    });
});
```

10.5　Discuz微社区

微社区是腾讯康盛Discuz！团队开发的手机上的新社区。

微社区是基于微信公众账号的互动社区，它可以广泛应用于微信服务号与订阅号，是微信公众账号运营者打造人气移动社区、增强用户黏性的有利工具。

微社区解决了同一微信公众账号下用户无法直接交流、互动的难题，把公众账号“一对多”的单向推送消息方式变成用户与用户、用户与平台之间的“多对多”沟通模式，双向交流给用户带来更好的互动体验，让互动更便捷、更畅快。

微社区首次把Web2.0的交互模式引入了微信公众平台，基于话题和共同兴趣，结合发帖和回复，促使用户从被动的消息接收者转向移动互联网消息创造者，在公众账号与用户，用户与用户之间的互动中共同完成内容制造和传播。

申请微社区时，打开http://wsq.qq.com/，在首页单击“立即开通”按钮，然后填写个人真实资料和微社区资料后，等待审核。

审核成功后，将获得一个社区的唯一地址，如方倍工作室的地址为http://wx.wsq.qq.com/182998484，将该地址填写到自定义菜单按钮或者通过自动回复带入链接中，用户即可跳转到微社区中。

方倍工作室的微社区如图10-11所示。

22:05
方倍社区
方倍社区
80话题
4782访问
方倍工作室
02-20
方倍社区是交流学习微信公众平台开发技术、运营等内容的社区平台。
21
分享
回复
韶华：很好！支持！
朱乾(颠覆营销)：↑
Thinking...：:(
发帖
1

图10-11　微社区

10.6　本章小结

本章介绍了一些开发实现技巧，包括表情飘落效果、回复QQ表情和Emoji表情、获得微信版本和手机操作系统、微信JS接口使用以及Discuz微社区。

附录A　关键词自动回复的规则

1.字数限制

微信公众平台认证与非认证用户的关键词自动回复设置规则上限为200条规则（每条规则名，最多可设置60个汉字），每条规则内最多设置10条关键词（每条关键词，最多可设置30个汉字）、5条回复（每条回复，最多可设置300个汉字）。

2.规则设置

您可通过微信公众平台设置多个关键词，如订阅用户发送的信息中含有您设置的关键词，则系统会自己回复。

同一规则中可设置5条回复内容，如您设置了“回复全部”，粉丝发送信息中含有您设置的关键词，会将您设置的多条回复会全部发送，若未设置“回复全部”，则会随机回复。

3.关键词设置

1）每个规则里可设置10个关键词，若设置了相同的关键词，但回复内容不同，系统会随机回复。

2）每个规则里可设置5条回复内容，若设置了多个回复内容（没有设置“回复全部”），系统会随机回复。

3）多条回复设置方法：进入编辑者模式，单击“关键词自动回复”→“添加规则”，输入关键词匹配内容后，再添加内容，然后选择“回复全部”即可。

4.匹配功能设置

1）若选择了全匹配，在编辑页面则会显示“已全匹配”：

对方发送的内容与设置的关键词须完全一样，才会触发关键词回复，不能多一个字符也不能少一个字符。比如设置“123”，仅回复“123”才会触发关键词回复。

2）若没有选择全匹配的情况下，编辑页面则会显示“未全匹配”：

只要对方发送内容包含设置的完整关键词，就会触发关键词回复给对方。比如设置“123”，回复“1234”会触发，但回复不完整的关键词“12”则不会触发关键词回复。

5.其他提示

1）文字中可以输入网页链接地址，但不支持设置超链接。

2）关注用户发送消息命中设置关键词回复规则后会有5秒钟的响应时间。

3）规则名称自己设定（作为区分作用）；

4）关键词配置（作为粉丝发来命中的关键词）；

5）回复信息（命中关键词后自动回复的信息内容）；

6）勾选了“回复全部”，只要粉丝命中关键词就会自动回复该规则内的所有回复；若未勾选，会随机回复；

7）建议在一个规则里设置一个关键词，以便粉丝获得想要的答案。

附录B　微信公众平台全局返回码说明

返回码	说明	返回码	说明
-1	系统繁忙	41002	缺少 appid 参数
0	请求成功	41003	缺少 refresh_token 参数
40001	获取 access_token 时 AppSecret 错误，或者 access_token 无效	41004	缺少 secret 参数
40002	不合法的凭证类型	41005	缺少多媒体文件数据
40003	不合法的 OpenID	41006	缺少 media_id 参数
40004	不合法的媒体文件类型	41007	缺少子菜单数据
40005	不合法的文件类型	41008	缺少 oauth code
40006	不合法的文件大小	41009	缺少 openid
40007	不合法的媒体文件 id	42001	access_token 超时
40008	不合法的消息类型	42002	refresh_token 超时
40009	不合法的图片文件大小	42003	oauth_code 超时
40010	不合法的语音文件大小	43001	需要 GET 请求
40011	不合法的视频文件大小	43002	需要 POST 请求
40012	不合法的缩略图文件大小	43003	需要 HTTPS 请求
40013	不合法的 APPID	43004	需要接收者关注
40014	不合法的 access_token	43005	需要好友关系
40015	不合法的菜单类型	44001	多媒体文件为空
40016	不合法的按钮个数	44002	POST 的数据包为空
40017	不合法的按钮个数	44003	图文消息内容为空
40018	不合法的按钮名字长度	44004	文本消息内容为空
40019	不合法的按钮 KEY 长度	45001	多媒体文件大小超过限制
40020	不合法的按钮 URL 长度	45002	消息内容超过限制
40021	不合法的菜单版本号	45003	标题字段超过限制
40022	不合法的子菜单级数	45004	描述字段超过限制
40023	不合法的子菜单按钮个数	45005	链接字段超过限制
40024	不合法的子菜单按钮类型	45006	图片链接字段超过限制
40025	不合法的子菜单按钮名字长度	45007	语音播放时间超过限制
40026	不合法的子菜单按钮 KEY 长度	45008	图文消息超过限制
40027	不合法的子菜单按钮 URL 长度	45009	接口调用超过限制
40028	不合法的自定义菜单使用用户	45010	创建菜单个数超过限制
40029	不合法的 oauth_code	45015	回复时间超过限制
40030	不合法的 refresh_token	45016	系统分组，不允许修改
40031	不合法的 openid 列表	45017	分组名字过长
40032	不合法的 openid 列表长度	45018	分组数量超过上限
40033	不合法的请求字符，不能包含 \uxxxx 格式的字符	46001	不存在媒体数据
40035	不合法的参数	46002	不存在的菜单版本
40038	不合法的请求格式	46003	不存在的菜单数据
40039	不合法的 URL 长度	46004	不存在的用户
40050	不合法的分组 id	47001	解析 JSON/XML 内容错误
40051	分组名字不合法	48001	api 功能未授权
41001	缺少 access_token 参数	50001	用户未授权该 api

附录C　SAE分词词性说明

代码	词性	代码	词性
0	不知道	99	机构团体名（“北大”）
10	形容词	100	其他专名（“专”的声母的第 1 个字母为 z，名词代码 n 和 z 并在一起。）
20	区别词	101	名处词
30	连词	102	地名（名处词专指：“中国”）
31	体词连接	103	n-m, 数词开头的名词（三个学生）
32	分句连接	104	n-rb, 以区别词 / 代词开头的名词（该学校，该生）
40	副词	107	拟声词
41	副词（“不”）	108	介词
42	副词（“没”）	110	量词
50	叹词	111	动量词（“趟”“遍”）
60	方位词	112	时间量词（“年”“月”“期”）
61	方位短语（处所词 + 方位词）	113	货币量词（“元”“美元”“英镑”）
62	方位短语（名词 + 方位词 “地上”）	120	代词
63	方位短语（动词 + 方位词 “取前”）	121	副词性代词（“怎么”）
64	方位短语（动词 + 方位词 “取前”）	122	数词性代词（“多少”）
70	前接成分	123	名词性代词（“什么”“谁”）
71	数词前缀（“数”——数十）	124	处所词性代词（“哪儿”）
72	时间词前缀（“公元”“明永乐”）	125	时间词性代词（“何时”）
73	姓氏	126	谓词性代词（“怎么样”）
74	姓氏	127	区别词性代词（“某”“每”）
80	后接成分	130	处所词（取英语 space 的第 1 个字母。“东部”）
81	数词后缀（“来”——十来个）	131	处所词（取英语 space 的第 1 个字母。“东部”）
82	时间词后缀（“初”“末”“时”）	132	时间词（取英语 time 的第 1 个字母）
83	名词后缀（“们”）	133	时间专指（“唐代”“西周”）
84	处所词后缀（“苑”“里”）	140	助词
85	状态词后缀（“然”）	141	定语助词（“的”）
86	状态词后缀（“然”）	142	状语助词（“地”）
87	状态词后缀（“然”）	143	补语助词（“得”）
90	数词	144	谓词后助词（“了、着、过”）
95	名词	145	体词后助词（“等、等等”）
96	人名（“毛泽东”）	146	助词（“所”）
97	机构团体（“团”的声母为 t，名词代码 n 和 t 并在一起。“公司”）	150	标点符号

（续）

代码	词性	代码	词性
151	顿号（“、”）	190	语素字
152	句号（“。”）	191	名词语素（“琥”）
153	分句尾标点（“，”“；”）	192	动词语素（“酊”）
154	搭配型标点左部	193	处所词语素（“中”“日”“美”）
155	搭配型标点右部（“》”“]”“)”）	194	时间词语素（“唐”“宋”“元”）
156	中缀型符号	195	状态词语素（“伟”“芳”）
160	语气词（取汉字“语”的声母。“吗”“吧”“啦”）	196	状态词语素（“伟”“芳”）
170	及物动词（取英语动词 verb 的第一个字母。）	200	不及物谓词（主谓结构“腰酸”“头疼”）
171	不及物谓词（谓宾结构“剃头”）	201	数量短语（“叁个”）
172	动补结构动词（“取出”“放到”）	202	代量短语（“这个”）
173	动词“是”	210	副形词（直接作状语的形容词）
174	动词“有”	211	名形词（具有名词功能的形容词）
175	趋向动词（“来”“去”“进来”）	212	副动词（直接作状语的动词）
176	助动词（“应该”“能够”）	213	名动词（指具有名词功能的动词）
180	状态词（不及物动词，v-o、sp 之外的不及物动词）	230	空格

附录D　微信公众平台基础接口PHP SDK

```php
<?php
/*
    方倍工作室
    CopyRight 2014 All Rights Reserved
*/

define("TOKEN", "weixin");

$wechatObj = new wechatCallbackapiTest();
if (!isset($_GET['echostr'])) {
    $wechatObj->responseMsg();
}else{
    $wechatObj->valid();
}

class wechatCallbackapiTest
{
    // 验证消息
    public function valid()
    {
        $echoStr = $_GET["echostr"];
        if($this->checkSignature()){
            echo $echoStr;
            exit;
        }
    }

    // 检查签名
    private function checkSignature()
    {
        $signature = $_GET["signature"];
        $timestamp = $_GET["timestamp"];
        $nonce = $_GET["nonce"];
        $token = TOKEN;
        $tmpArr = array($token, $timestamp, $nonce);
        sort($tmpArr, SORT_STRING);
        $tmpStr = implode($tmpArr);
        $tmpStr = sha1($tmpStr);

        if($tmpStr == $signature){
            return true;
        }else{
            return false;
        }
    }

    // 响应消息
    public function responseMsg()
    {
        $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
        if (!empty($postStr)){
            $this->logger("R ".$postStr);
            $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
            $RX_TYPE = trim($postObj->MsgType);

            switch ($RX_TYPE)
            {
                case "event":
                    $result = $this->receiveEvent($postObj);
                    break;
                case "text":
                    $result = $this->receiveText($postObj);
                    break;
                case "image":
```

```
                    $result = $this->receiveImage($postObj);
                    break;
                case "location":
                    $result = $this->receiveLocation($postObj);
                    break;
                case "voice":
                    $result = $this->receiveVoice($postObj);
                    break;
                case "video":
                    $result = $this->receiveVideo($postObj);
                    break;
                case "link":
                    $result = $this->receiveLink($postObj);
                    break;
                default:
                    $result = "unknow msg type: ".$RX_TYPE;
                    break;
            }
            $this->logger("T ".$result);
            echo $result;
        }else {
            echo "";
            exit;
        }
    }

    // 接收事件消息
    private function receiveEvent($object)
    {
        $content = "";
        switch ($object->Event)
        {
            case "subscribe":
                $content = "欢迎关注方倍工作室 ";
                $content .= (!empty($object->EventKey))?("\n来自二维码场景 ".str_replace("qrscene_",",",$object->EventKey)):"";
                break;
            case "unsubscribe":
                $content = "取消关注";
                break;
            case "SCAN":
                $content = "扫描场景 ".$object->EventKey;
                break;
            case "CLICK":
                switch ($object->EventKey)
                {
                    case "COMPANY":
                        $content = "方倍工作室提供互联网相关产品与服务。";
                        break;
                    default:
                        $content = "点击菜单:".$object->EventKey;
                        break;
                }
                break;
            case "LOCATION":
               $content = "上传位置:纬度 ".$object->Latitude.";经度".$object->Longitude;
                break;
            default:
                $content = "receive a new event: ".$object->Event;
                break;
        }
        $result = $this->transmitText($object, $content);
        return $result;
    }

    // 接收文本消息
    private function receiveText($object)
    {
        $keyword = trim($object->Content);
        $content = "当前时间:".date("Y-m-d H:i:s",time());

        if(is_array($content)){
            if (isset($content[0]['PicUrl'])){
                $result = $this->transmitNews($object, $content);
            }else if (isset($content['MusicUrl'])){
                $result = $this->transmitMusic($object, $content);
            }
        }else{
            $result = $this->transmitText($object, $content);
        }
        return $result;
    }

    // 接收图片消息
    private function receiveImage($object)
    {
        $content = array("MediaId"=>$object->MediaId);
        $result = $this->transmitImage($object, $content);
        return $result;
    }

    // 接收位置消息
    private function receiveLocation($object)
    {
        $content = "你发送的是位置,纬度为:".$object->Location_X."；经度为:".$object->Location_Y."；缩放级别为:".$object->Scale."；位置为:".$object->Label;
        $result = $this->transmitText($object, $content);
        return $result;
    }

    // 接收语音消息
    private function receiveVoice($object)
    {
        if (empty($object->Recognition)){
            $content = array("MediaId"=>$object->MediaId);
            $result = $this->transmitVoice($object, $content);
        }else{
            $content = "你刚才说的是:".$object->Recognition;
            $result = $this->transmitText($object, $content);
        }

        return $result;
    }

    // 接收视频消息
    private function receiveVideo($object)
    {
        $content = array("MediaId"=>$object->MediaId, "ThumbMediaId"=>$object->
ThumbMediaId, "Title"=>"", "Description"=>"");
        $result = $this->transmitVideo($object, $content);
        return $result;
    }

    // 接收链接消息
    private function receiveLink($object)
    {
        $content = "你发送的是链接,标题为:".$object->Title."；内容为:".$object->Description."；链接地址为:".$object->Url;
        $result = $this->transmitText($object, $content);
        return $result;
    }

    // 发送文本消息
    private function transmitText($object, $content)
    {
        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[text]]></MsgType>
<Content><![CDATA[%s]]></Content>
</xml>";
```

```
        $result = sprintf($textTpl, $object->FromUserName, $object->ToUserName, time(), $content);
        return $result;
    }

    // 发送图片消息
    private function transmitImage($object, $imageArray)
    {
        $itemTpl = "<Image>
    <MediaId><![CDATA[%s]]></MediaId>
</Image>";

        $item_str = sprintf($itemTpl, $imageArray['MediaId']);

        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[image]]></MsgType>
$item_str
</xml>";

        $result = sprintf($textTpl, $object->FromUserName, $object->ToUserName, time());
        return $result;
    }

    // 发送语音消息
    private function transmitVoice($object, $voiceArray)
    {
        $itemTpl = "<Voice>
    <MediaId><![CDATA[%s]]></MediaId>
</Voice>";

        $item_str = sprintf($itemTpl, $voiceArray['MediaId']);

        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[voice]]></MsgType>
$item_str
</xml>";

       $result = sprintf($textTpl, $object->FromUserName, $object->ToUserName, time());
        return $result;
    }

    // 发送视频消息
    private function transmitVideo($object, $videoArray)
    {
        $itemTpl = "<Video>
    <MediaId><![CDATA[%s]]></MediaId>
    <ThumbMediaId><![CDATA[%s]]></ThumbMediaId>
    <Title><![CDATA[%s]]></Title>
    <Description><![CDATA[%s]]></Description>
</Video>";

       $item_str = sprintf($itemTpl, $videoArray['MediaId'], $videoArray['ThumbMediaId'], $videoArray['Title'], $videoArray['Description']);

        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[video]]></MsgType>
$item_str
</xml>";

        $result = sprintf($textTpl, $object->FromUserName, $object->ToUserName, time());
        return $result;
    }

    // 发送图文消息
    private function transmitNews($object, $newsArray)
    {
        if(!is_array($newsArray)){
            return;
        }
        $itemTpl = "    <item>
        <Title><![CDATA[%s]]></Title>
        <Description><![CDATA[%s]]></Description>
        <PicUrl><![CDATA[%s]]></PicUrl>
        <Url><![CDATA[%s]]></Url>
    </item>
";
        $item_str = "";
        foreach ($newsArray as $item){
           $item_str .= sprintf($itemTpl, $item['Title'], $item['Description'], $item['PicUrl'], $item['Url']);
        }
        $newsTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[news]]></MsgType>
<Content><![CDATA[]]></Content>
<ArticleCount>%s</ArticleCount>
<Articles>
$item_str</Articles>
</xml>";

        $result = sprintf($newsTpl, $object->FromUserName, $object->ToUserName, time(), count($newsArray));
        return $result;
    }

    // 发送音乐消息
    private function transmitMusic($object, $musicArray)
    {
        $itemTpl = "<Music>
    <Title><![CDATA[%s]]></Title>
    <Description><![CDATA[%s]]></Description>
    <MusicUrl><![CDATA[%s]]></MusicUrl>
    <HQMusicUrl><![CDATA[%s]]></HQMusicUrl>
</Music>";

        $item_str = sprintf($itemTpl, $musicArray['Title'], $musicArray['Description'], $musicArray['MusicUrl'], $musicArray['HQMusicUrl']);

        $textTpl = "<xml>
<ToUserName><![CDATA[%s]]></ToUserName>
<FromUserName><![CDATA[%s]]></FromUserName>
<CreateTime>%s</CreateTime>
<MsgType><![CDATA[music]]></MsgType>
$item_str
</xml>";

        $result = sprintf($textTpl, $object->FromUserName, $object->ToUserName, time());
       return $result;
    }

    // 日志记录
    private function logger($log_content)
    {
        if(isset($_SERVER['HTTP_APPNAME'])){   // SAE
            sae_set_display_errors(false);
            sae_debug($log_content);
            sae_set_display_errors(true);
        }else if($_SERVER['REMOTE_ADDR'] != "127.0.0.1"){ // LOCAL
```

```
            $max_size = 10000;
            $log_filename = "log.xml";
            if(file_exists($log_filename) and (abs(filesize($log_filename))> $max_size)){unlink($log_filename);}
            file_put_contents($log_filename, date('H:i:s')." ".$log_content."\r\n", FILE_APPEND);
        }
    }
}
?>
```

附录E　微信公众平台自定义菜单及高级接口PHP SDK

```
/*
    方倍工作室
    CopyRight 2014 All Rights Reserved
*/

class class_weixin
{
    var $appid = "";
    var $appsecret = "";

    // 构造函数,获取Access Token
    public function __construct($appid = NULL, $appsecret = NULL)
    {
        if($appid){
            $this->appid = $appid;
        }
        if($appsecret){
            $this->appsecret = $appsecret;
        }

        // hardcode
        $this->lasttime = 1395049256;
        $this->access_token = "nRZvVpDU7LxcSi7GnG2LrUcmKbAECzRf0NyDBwKlng4nMPf88d34pkzdNcvhqm4clidLGAS18cN1RTSK60p49zIZY4aO13sF-eqsCs0xjlbad-lKVskk8T7gALQ5dIrgXbQQ_TAesSasjJ210vIqTQ";

        if (time() > ($this->lasttime + 7200)){
            $url = "https:// api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=".$this->appid."&secret=".$this->appsecret;
            $res = $this->https_request($url);
            $result = json_decode($res, true);
            // save to Database or Memcache
            $this->access_token = $result["access_token"];
            $this->lasttime = time();
        }
    }

    // 获取关注者列表
    public function get_user_list($next_openid = NULL)
    {
        $url = "https://api.weixin.qq.com/cgi-bin/user/get?access_token=".$this->access_token."&next_openid=".$next_openid;
        $res = $this->https_request($url);
        return json_decode($res, true);
    }

    // 获取用户基本信息
    public function get_user_info($openid)
    {
        $url = "https:// api.weixin.qq.com/cgi-bin/user/info?access_token=".$this->access_token."&openid=".$openid."&lang=zh_CN";
        $res = $this->https_request($url);
        return json_decode($res, true);
    }

    // 创建菜单
    public function create_menu($data)
    {
        $url = "https:// api.weixin.qq.com/cgi-bin/menu/create?access_token=".$this->access_token;
        $res = $this->https_request($url, $data);
        return json_decode($res, true);
    }

    // 发送客服消息,已实现发送文本,其他类型可扩展
    public function send_custom_message($touser, $type, $data)
    {
        $msg = array('touser' =>$touser);
        switch($type)
        {
            case 'text':
                $msg['msgtype'] = 'text';
                $msg['text']    = array('content'=> urlencode($data));
                break;
        }
        $url = "https:// api.weixin.qq.com/cgi-bin/message/custom/send?access_token=".$this->access_token;
        return $this->https_request($url, urldecode(json_encode($msg)));
    }

    // 生成参数二维码
    public function create_qrcode($scene_type, $scene_id)
    {
        switch($scene_type)
        {
            case 'QR_LIMIT_SCENE': // 永久
                $data = '{"action_name": "QR_LIMIT_SCENE", "action_info": {"scene": {"scene_id": '.$scene_id.'}}}';
                break;
            case 'QR_SCENE':       // 临时
                $data = '{"expire_seconds": 1800, "action_name": "QR_SCENE", "action_info": {"scene": {"scene_id": '.$scene_id.'}}}';
              break;
        }
        $url = "https:// api.weixin.qq.com/cgi-bin/qrcode/create?access_token=".$this->access_token;
        $res = $this->https_request($url, $data);
        $result = json_decode($res, true);
        return "https:// mp.weixin.qq.com/cgi-bin/showqrcode?ticket=".urlencode($result["ticket"]);
    }

    // 创建分组
    public function create_group($name)
    {
        $data = '{"group": {"name": "'.$name.'"}}';
        $url = "https:// api.weixin.qq.com/cgi-bin/groups/create?access_token=".$this->access_token;
        $res = $this->https_request($url, $data);
        return json_decode($res, true);
    }

    // 移动用户分组
    public function update_group($openid, $to_groupid)
```

```
    {
        $data = '{"openid":"'.$openid.'","to_groupid":’.$to_groupid.’}’;
        $url = "https:// api.weixin.qq.com/cgi-bin/groups/members/update?access_token=".$this->access_token;
        $res = $this->https_request($url, $data);
        return json_decode($res, true);
    }

    // 上传多媒体文件
    public function upload_media($type, $file)
    {
        $data = array("media"  => "@".dirname(__FILE__).’\\’.$file);
        $url = "http://file.api.weixin.qq.com/cgi-bin/media/upload?access_token=".$this->access_token."&type=".$type;
        $res = $this->https_request($url, $data);
        return json_decode($res, true);
    }

    // 地理位置逆解析
    public function location_geocoder($latitude, $longitude)
    {
        $url = "http://api.map.baidu.com/geocoder/v2/?ak=B944e1fce373e33ea4627f95f54f2ef9&location=".$latitude.",".$longitude."&coordtype=gcj02ll&output=json";
        $res = $this->https_request($url);
        $result = json_decode($res, true);
        return $result["result"]["addressComponent"];
    }

    // https请求（支持GET和POST）
    protected function https_request($url, $data = null)
    {
        $curl = curl_init();
        curl_setopt($curl, CURLOPT_URL, $url);
        curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
        curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, FALSE);
        if (!empty($data)){
            curl_setopt($curl, CURLOPT_POST, 1);
            curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
        }
        curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
        $output = curl_exec($curl);
        curl_close($curl);
        return $output;
    }
}
```